PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : G06F 1/00 — A2

(11) International Publication Number: **WO 99/57623**

(43) International Publication Date: 11 November 1999 (11.11.99)

(21) International Application Number: PCT/IB99/01250

(22) International Filing Date: 29 April 1999 (29.04.99)

(30) Priority Data:
09/070,597 30 April 1998 (30.04.98) US

(71) Applicant: FRAUNHOFER CENTER FOR RESEARCH IN COMPUTER GRAPHICS, INC. [US/US]; Suite 2, 321 S. Main Street, Providence, RI 02903 (US).

(72) Inventor: ZHAO, Jian; 64 Thomas Olney Common, Providence, RI 02904 (US).

(81) Designated States: JP, KR, European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

Published
Without international search report and to be republished upon receipt of that report.

(54) Title: ACTIVE WATERMARKS AND WATERMARK AGENTS

(57) Abstract

Techniques for protecting the security of digital representations, and of analog forms made from them. The techniques include authentication techniques that can authenticate both a digital representation and an analog form produced from the digital representation, an active watermark that contains program code that may be executed when the watermark is read, and a watermark agent that reads watermarks and sends messages with information concerning the digital representations that contain the watermarks. The authentication techniques use semantic information to produce authentication information. Both the semantic information and the authentication information survive when an analog form is produced from the digital representation. In one embodiment, the semantic information is alphanumeric characters and the authentication information is either contained in a watermark embedded in the digital representation or expressed as a bar code. With the active watermark, the watermark includes program code. When a watermark reader reads the watermark, it may cause the program code to be executed. One application of active watermarks is making documents that send messages when they are operated on. A watermark agent may be either a permanent resident of a node in a network or of a device such as a copier or it may move from one network node to another. In the device or node, the watermark agent executes code which examines digital representations residing in the node or device for watermarked digital representations that are of interest to the watermark agent. The watermark agent then sends messages which report the results of its examination of the digital representations. If the watermarks are active, the agent and the active watermark may cooperate and the agent may cause some or all of the code that an active watermark contains to be executed.

BNSDOCID: <WO___9957623A2_I_>

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	**ES**	Spain	**LS**	Lesotho	**SI**	Slovenia
AM	Armenia	**FI**	Finland	**LT**	Lithuania	**SK**	Slovakia
AT	Austria	**FR**	France	**LU**	Luxembourg	**SN**	Senegal
AU	Australia	**GA**	Gabon	**LV**	Latvia	**SZ**	Swaziland
AZ	Azerbaijan	**GB**	United Kingdom	**MC**	Monaco	**TD**	Chad
BA	Bosnia and Herzegovina	**GE**	Georgia	**MD**	Republic of Moldova	**TG**	Togo
BB	Barbados	**GH**	Ghana	**MG**	Madagascar	**TJ**	Tajikistan
BE	Belgium	**GN**	Guinea	**MK**	The former Yugoslav Republic of Macedonia	**TM**	Turkmenistan
BF	Burkina Faso	**GR**	Greece			**TR**	Turkey
BG	Bulgaria	**HU**	Hungary	**ML**	Mali	**TT**	Trinidad and Tobago
BJ	Benin	**IE**	Ireland	**MN**	Mongolia	**UA**	Ukraine
BR	Brazil	**IL**	Israel	**MR**	Mauritania	**UG**	Uganda
BY	Belarus	**IS**	Iceland	**MW**	Malawi	**US**	United States of America
CA	Canada	**IT**	Italy	**MX**	Mexico	**UZ**	Uzbekistan
CF	Central African Republic	**JP**	Japan	**NE**	Niger	**VN**	Viet Nam
CG	Congo	**KE**	Kenya	**NL**	Netherlands	**YU**	Yugoslavia
CH	Switzerland	**KG**	Kyrgyzstan	**NO**	Norway	**ZW**	Zimbabwe
CI	Côte d'Ivoire	**KP**	Democratic People's Republic of Korea	**NZ**	New Zealand		
CM	Cameroon			**PL**	Poland		
CN	China	**KR**	Republic of Korea	**PT**	Portugal		
CU	Cuba	**KZ**	Kazakstan	**RO**	Romania		
CZ	Czech Republic	**LC**	Saint Lucia	**RU**	Russian Federation		
DE	Germany	**LI**	Liechtenstein	**SD**	Sudan		
DK	Denmark	**LK**	Sri Lanka	**SE**	Sweden		
EE	Estonia	**LR**	Liberia	**SG**	Singapore		

BNSDOCID: <WO 9957623A2 I >

Active Watermarks and Watermark Agents

Cross Reference to Related Patent Applications

This application has the same *Detailed Description* as Jian Zhao and E. Koch, *Digital Authentication with Analog Documents*, assigned to Mediasec LLP and filed on even date with this application.

Background of the Invention

1. Field of the Invention

The invention relates generally to digital representations of images and other information and more specifically to techniques for protecting the security of digital representations and of analog forms produced from them.

2. Description of the Prior Art

Nowadays, the easiest way to work with pictures or sounds is often to make digital representations of them. Once the digital representation is made, anyone with a computer can copy the digital representation without degradation, can manipulate it, and can send it virtually instantaneously to anywhere in the world. The Internet, finally, has made it possible for anyone to distribute any digital representation from anywhere in the world

From the point of view of the owners of the digital representations, there is one problem with all of this: pirates, too, have computers, and they can use them to copy, manipulate, and distribute digital representations as easily as the legitimate owners and users can. If the owners of the original digital representations are to be properly compensated for making or publishing them, the digital representations must be protected from pirates. There are a number of different approaches that can be used:

- the digital representation may be rendered unreadable except by its intended recipients; this is done with encryption techniques;

BNSDOCID: <WO___9957623A2_I_>

- the digital representation may be marked to indicate its authenticity; this is done with digital signatures;
- the digital representation may contain information from which it may be determined whether it has been tampered with in transit; this information is termed a *digest* and the digital signature often includes a digest;
- the digital representation may contain a *watermark*, an invisible indication of ownership which cannot be removed from the digital representation and may even be detected in an analog copy made from the digital representation; and
- the above techniques can be employed in systems that not only protect the digital representations, but also meter their use and/or detect illegal use.

For an example of a system that uses encryption to protect digital representations, see U.S. Patent 5,646,999, Saito, *Data Copyright Management Method*, issued July 8, 1997; for a general discussion of digital watermarking, see Jian Zhao, "Look, It's Not There", in: *BYTE Magazine*, January, 1997. Detailed discussions of particular techniques for digital watermarking may be found in E. Koch and J. Zhao, "Towards Robust and Hidden Image Copyright Labeling", in: *Proc. Of 1995 IEEE Workshop on Nonlinear Signal and Image Processing*, June 20-22, 1995, and in U.S. Pat. No. 5,710,834, Rhoads, *Method and Apparatus Responsive to a Code Signal Conveyed through a Graphic Image*, issued Jan. 20, 1998. For an example of a commercial watermarking system that uses the digital watermarking techniques disclosed in the Rhoads patent, see *Digimarc Watermarking Guide*, Digimarc Corporation, 1997, available at in March, 1998 at `http://www.digimarc.com`.

FIG. 1 shows a prior-art system 101 which employs the above protection techniques. A number of digital representation clients 105, of which only one, digital representation client 105(j) is shown, are connected via a network 103 such as the Internet to a digital representation server 129 which receives digital representations from clients 105 and distributes them to clients 105. Server 129 includes a data storage device 133 which contains copied digital representations 135 for distribution and a management data base 139. Server

BNSDOCID: <WO___9957623A2_I_>

129 further includes a program for managing the digital representations 135, a program for reading and writing watermarks 109, a program for authenticating a digital representation and confirming that a digital representation is authentic 111, and a program for encrypting and decrypting digital representations 113. Programs 109, 111, and 113 together make up security programs 107.

Client 105 has its own versions of security programs 107; it further has editor/viewer program 115 which lets the user of client 105 edit and/or view digital representations that it receives via network 103 or that are stored in storage device 117. Storage device 117 as shown contains an original digital representation 119 which was made by a user of client 105 and a copied digital representation 121 that was received from DR Server 129. Of course, the user may have made original representation 119 by modifying a copied digital representation. Editor/viewer program 115, finally, permits the user to output digital representations to analog output devices 123. Included among these devices are a display 123, upon which an analog image 124 made from a digital representation may be displayed and a printer 127 upon which an analog image 126 made from the digital representation may be printed. A loudspeaker may also be included in analog output devices 123. The output of the analog output device will be termed herein an *analog form* of the digital representation. For example, if the output device is a printer, the analog form is printed sheet 126; if it is a display device, it is display 124.

When client 105(j) wishes to receive a digital representation from server 129, it sends a message requesting the the digital representation to server 129. The message includes at least an identification of the desired digital representation and an identification of the user. Manager 131 responds to the request by locating the digital representation in CDRs 135, consulting management data base 139 to determine the conditions under which the digital representation may be distributed and the status of the user of client 105 as a customer. If the information in data base 139 indicates to manager 131 that the transaction should go forward, manager 131 sends client 105(j) a copy of the selected digital representation. In the

BNSDOCID: <WO___9957623A2_I_>

course of sending the copy, manager 131 may use watermark reader/writer 109 to add a watermark to the digital representation, use authenticator/confirmer 111 to add authentication information, and encrypter/decrypter 113 to encrypt the digital representation in such a fashion that it can only be decrypted in DR client 105(j).

When client 105(j) receives the digital representation, it decrypts it using program 113, confirms that the digital representation is authentic using program 111, and editor/viewer 115 may use program 109 to display the watermark. The user of client 105(j) may save the encrypted or unencrypted digital representation in storage 117. The user of client 105(j) may finally employ editor/viewer 115 to decode the digital representation and output the results of the decoding to an analog output device 123. Analog output device 123 may be a display device 125, a printer 127, or in the case of digital representations of audio, a loudspeaker.

It should be pointed out that when the digital representation is displayed or printed in analog form, the only remaining protection against copying is watermark 128, which cannot be perceived in the analog form by the human observer, but which can be detected by scanning the analog form and using a computer to find watermark 128. Watermark 128 thus provides a backup to encryption: if a digital representation is pirated, either because someone has broken the encryption, or more likely because someone with legitimate access to the digital representation has made illegitimate copies, the watermark at least makes it possible to determine the owner of the original digital representation and given that evidence, to pursue the pirate for copyright infringement and/or violation of a confidentiality agreement.

If the user of client 105(j) wishes to send an original digital representation 119 to DR server 129 for distribution, editor/viewer 115 will send digital representation 119 to server 129. In so doing, editor/viewer 115 may use security programs 107 to watermark the digital representation, authenticate it, and encrypt it so that it can be decrypted only by DR Server 129. Manager 131 in DR server 129 will, when it receives digital representation 119, use

BNSDOCID: <WO___9957623A2_I_>

security programs 107 to decrypt digital representation 119, confirm its authenticity, enter information about it in management data base 139, and store it in storage 133.

In the case of the Digimarc system referred to above, manager 131 also includes a World Wide Web *spider*, that is, a program that systematically follows World Wide Web links such as HTTP and FTP links and fetches the material pointed to by the links.

Manager program 131 uses watermark reading/writing program to read any watermark, and if the watermark is known to management database 139, manager program 131 takes whatever action may be required, for example, determining whether the site from which the digital representation was obtained has the right to have it, and if not, notifying the owner of the digital representation.

While encryption, authentication, and watermarking have made it much easier for owners of digital representations to protect their property, problems still remain. One such problem is that the techniques presently used to authenticate digital documents do not work with analog forms; consequently, when the digital representation is output in analog form, the authentication is lost. Another is that present-day systems for managing digital representations are not flexible enough. A third is that watermark checking such as that done by the watermark spider described above is limited to digital representations available on the Internet. It is an object of the present invention to overcome the above problems and thereby to provide improved techniques for distributing digital representations.

Summary of the Invention

In one aspect, the invention is an active watermark, that is, a watermark in which the information included in the watermark includes program code that can be executed when the watermark is read. What the program code does is of course completely arbitrary. For example, the code in the active watermark can send a message each time a particular operation is performed on the digital representation containing the active watermark. One

BNSDOCID: <WO___9957623A2_I_>

use for such an active watermark is for billing: each time a digital representation with an active watermark is copied, for instance, the digital representation may send a message to a billing server. Another use is destroying the digital representation if a user attempts an operation for which the user has no privileges. In this aspect, the invention includes apparatus and methods for making and reading active watermarks. The methods and apparatus for making active watermarks may be employed anywhere present-day watermark makers are employed, and the methods and apparatus for reading active watermarks may be employed anywhere present-day watermark readers are employed.

In another aspect, the invention is a watermark agent which is located in a device upon which digital representations containing watermarks are resident. The watermark agent reads the watermark in a digital representation and performs actions ranging from sending a message to the user through sending a message to a monitoring agent, moving the digital representation, or changing its access rights to destroying the digital representation. Some watermark agents are mobile. A mobile watermark agent moves from node to node in a network. In each node, it examines the watermarks on digital representations stored in the node and sends messages reporting its findings to a monitoring agent located in the network. When a watermark agent encounters a digital representation with an active watermark, the watermark agent may execute the program code contained in the active watermark.

Other objects and advantages of the invention will be apparent to those skilled in the arts to which the invention pertains upon perusing the following *Detailed Description* and Drawing, wherein:

Brief Description of the Drawing

FIG. 1 is a block diagram of a prior-art system for securely distributing digital representations;

FIG. 2 is a diagram of a first embodiment of an analog form that can be authenticated;

FIG. 3 is a diagram of a second embodiment of an analog form that can be authenticated;

BNSDOCID: <WO___9957623A2_I_>

FIG. 4 is a diagram of a system for adding authentication information to an analog form;

FIG. 5 is a diagram of a system for authenticating an analog form;

FIG. 6 is a diagram of a system for making an active watermark;

FIG. 7 is an example of code from an active watermark;

FIG. 8 is a diagram of a system for executing the code in an active watermark;

FIG. 9 is a diagram of a system for producing a watermark agent;

FIG. 10 is a diagram of a system for receiving a watermark agent;

FIG. 11 is a detailed diagram of access information 603; and

FIG. 12 is an example of code executed by a watermark agent.

The reference numbers in the drawings have at least three digits. The two rightmost digits are reference numbers within a figure; the digits to the left of those digits are the number of the figure in which the item identified by the reference number first appears. For example, an item with reference number 203 first appears in FIG. 2.

Detailed Description

The following *Detailed Description* will first disclose a technique for authenticating digital representations that survives output of an analog form of the digital representation, will then disclose *active watermarks*, that is, watermarks that contain programs, and will finally disclose *watermark agents*, that is, programs which examine the digital watermarks on digital representations stored in a system and thereby locate digital representations that are being used improperly.

Authentication that is preserved in analog forms: FIGs. 2-5

Digital representations are *authenticated* to make sure that they have not been altered in transit. Alteration can occur as a result of transmission errors that occur during the course of transmission from the source of the digital representation to its destination, as a result of errors that arise due to damage to the storage device being used to transport the digital representation, as a result of errors that arise in the course of writing the digital representation to the storage device or reading the digital representation from the storage

BNSDOCID: <WO___9957623A2_I_>

device, or as a result of human intervention. A standard technique for authentication is to make a *digest* of the digital representation and send the digest to the destination together with the digital representation. At the destination, another digest is made from the digital representation as received and compared with the first. If they are the same, the digital representation has not changed. The digest is simply a value which is much shorter than the digital representation but is related to it such that any change in the digital representation will with very high probability result in a change to the digest.

Where human intervention is a serious concern, the digest is made using a *one-way hash function*, that is, a function that produces a digest from which it is extremely difficult or impossible to learn anything about the input that produced it. The digest may additionally be encrypted so that only the recipient of the digital representation can read it. A common technique is to use the encrypted digest as the *digital signature* for the digital representation, that is, not only to show that the digital representation has not been altered in transit, but also to show that it is from whom it purports to be from. If the sender and the recipient have exchanged public keys, the sender can make the digital signature by encrypting the digest with the sender's private key. The recipient can use the sender's public key to decrypt the digest, and having done that, the recipient compares the digest with the digest made from the received digital representation. If they are not the same, either the digital representation has been altered or the digital representation is not from the person to whom the public key used to decrypt the digest belongs. For details on authentication, see Section 3.2 of Bruce Schneier, *Applied Cryptography*, John Wiley and Sons, 1994.

The only problem with authentication is that it is based entirely on the digital representation. The information used to make the digest is lost when the digital representation is output in analog form. For example, if the digital representation is a document, there is no way of determining from a paper copy made from the digital representation whether the digital representation from which the paper copy was made is authentic or whether the paper copy is itself a true copy of the digital representation.

BNSDOCID: <WO___9957623A2_I_>

While digital watermarks survive and remain detectable when a digital representation is output in analog form, the authentication problem cannot be solved simply by embedding the digest or digital signature in the watermark. There are two reasons for this:

- Watermarking changes the digital representation; consequently, if a digital representation is watermarked after the original digest is made, the watermarking invalidates the original digest, i.e., it is no longer comparable with the new digest that the recipient makes from the watermarked document.
- More troublesome still, when a digital representation is output in analog form, so much information about the digital representation is lost that the digital representation cannot be reconstructed from the analog form. Thus, even if the original digest is still valid, there is no way of producing a comparable new digest from the analog form.

What is needed to overcome these problems is an authentication technique which uses information for authentication which is independent of the particular form of the digital representation and which will be included in the analog form when the analog form is output. As will be explained in more detail in the following, the first requirement is met by selecting *semantic information* from the digital representation and using only the semantic information to make the digest. The second requirement is met by incorporating the digest into the digital representation in a fashion such that it on the one hand does not affect the semantic information used to make the digest and on the other hand survives in the analog form. In the case of documents, an authentication technique which meets these requirements can be used not only to authenticate analog forms of documents that exist primarily in digital form, but also to authenticate documents that exist primarily or only in analog form, for example paper checks and identification cards.

Semantic information

The semantic information in a digital representation is that portion of the information in the digital representation that *must be present* in the analog form made from the digital representation if the human who perceive the analog form is to consider it a copy of the

BNSDOCID: <WO___9957623A2_I_>

original from which the digital representation was made. For example, the semantic information in a digital representation of an image of a document is the representations of the alphanumeric characters in the document, where *alphanumeric* is understood to include representations of any kind of written characters or punctuation marks, including those belonging to non-Latin alphabets, to syllabic writing systems, and to ideographic writing systems. Given the alphanumeric characters, the human recipient of the analog form can determine whether a document is a copy of the original, even though the characters may have different fonts and may have been formatted differently in the original document. There is analogous semantic information in digital representations of pictures and of audio information. In the case of pictures, it is the information that is required for the human that perceives the analog form to agree that the analog form is a copy (albeit a bad one) of the original picture, and the same is the case with audio information.

In the case of a document written in English, the semantic information in the document is the letters and punctuation of the document. If the document is in digital form, it may be represented either as a digital image or in a text representation language such as those used for word processing or printing. In the first case, optical character recognition (OCR) technology may be applied to the image to obtain the letters and punctuation; in the second case, the digital representation may be parsed for the codes that are used to represent the letters and punctuation in the text representation language. If the document is in analog form, it may be scanned to produce a digital image and the OCR technology applied to the digital image produced by scanning.

Using semantic information to authenticate an analog form: FIGs. 2 and 3

Because the semantic information *must be present* in the analog form, it may be read from the analog form and used to compute a new digest. If the old digest was similarly made from the semantic information in the digital representation and the old digest is readable from the analog form, the new digest and the old digest can be compared as described in the discussion of authentication above to determine the authenticity of the analog form.

BNSDOCID: <WO 9957623A2 I >

FIG. 2 shows one technique 201 for incorporating the old digest into an analog form 203. Analog form 203 of course includes semantic information 205; here, analog form 203 is a printed or faxed document and semantic information 205 is part or all of the alphanumeric characters on analog form 203. Sometime before analog form 203 was produced, semantic information 205 in the digital representation from which analog form 203 was produced was used to make semantic digest 207, which was incorporated into analog form 203 at a location which did not contain semantic information 205 when analog form 203 was printed. In some embodiments, semantic digest 207 may be added to the original digital representation; in others, it may be added just prior to production of the analog form. Any representation of semantic digest 207 which is detectable from analog form 203 may be employed; in technique 201, semantic digest 207 is a visible bar code. Of course, semantic digest 207 may include additional information; for example, it may be encrypted as described above and semantic digest 207 may include an identifier for the user whose public key is required to decrypt semantic digest 207. In such a case, semantic digest 207 is a digital signature that persists in the analog form.

With watermarking, the semantic digest can be invisibly added to the analog form. This is shown in FIG. 3. In technique 301, analog form 303 again includes semantic information 305. Prior to producing analog form 303, the semantic information in the digital representation from which analog form 303 is produced is used as described above to produce semantic digest 207; this time, however, semantic digest 207 is incorporated into watermark 307, which is added to the digital representation before the analog form is produced from the digital representation and which, like the bar code of FIG. 2, survives production of the analog form. A watermark reader can read watermark 307 from a digital image made by scanning analog form 303, and can thereby recover semantic digest 207 from watermark 307. As was the case with the visible semantic digest, the semantic digest in watermark 307 may be encrypted and may also function as a digital signature.

Adding a semantic digest to an analog form: FIG. 4

BNSDOCID: <WO___9957623A2_I_>

FIG. 4 shows a system 401 for adding a semantic digest to an analog form 203. The process begins with digital representation 403, whose contents include semantic information 205. Digital representation 403 is received by semantics reader 405, which reads semantic information 205 from digital representation 403. Semantics reader 405's operation will depend on the form of the semantic information. For example, if digital representation 403 represents a document, the form of the semantic information will depend on how the document is represented. If it is represented as a bit-map image, the semantic information will be images of alphanumeric characters in the bit map; if it is represented using one of the many representations of documents that express alphanumeric characters as codes, the semantic information will be the codes for the alphanumeric characters. In the first case, semantics reader 405 will be an optical character reading (OCR) device; in the second, it will simply parse the document representation looking for character codes.

In any case, at the end of the process, semantics reader 405 will have extracted some form of semantic information, for example the ASCII codes corresponding to the alphanumeric characters, from representation 403. This digital information is then provided to digest maker 409, which uses it to make semantic digest 411 in any of many known ways. Depending on the kind of document the semantic digest is made from and its intended use, the semantic digest may have a form which requires an exact match with the new digest or may have a form which permits a "fuzzy" match. Digital representation 403 and semantic digest 411 are then provided to digest incorporator 413, which incorporates a representation 207 of digest 411 into the digital representation used to produce analog form 203. As indicated above, the representation must be incorporated in such a way that it does not affect semantic information 205. Incorporator 413 then outputs the representation it produces to analog form producer 415, which produces analog form 203 in the usual fashion. Analog form 203 of course includes semantic information 205 and representation 207 of semantic digest 411. Here, the bar code is used, but representation 207 could equally be part of a watermark, as in analog form 303. Components 405, 409, and 413 may be implemented as programs executed on a digital computer system; analog form producer 415 may be any device which can output an

BNSDOCID: <WO___9957623A2_I_>

analog form.

Authenticating an analog form that has a semantic digest

FIG. 5 shows a system 501 for authenticating an analog form 503 that has a semantic digest 207. Analog form 503 is first provided to semantic digest reader 505 and to semantics reader 505. Semantic digest reader 505 reads semantic digest 207; if semantic digest 207 is a bar code, semantic digest reader 505 is a bar code reader; if semantic digest 207 is included in a digital watermark, semantic digest reader 505 is a digital watermark reader which receives its input from a scanner. If semantic digest 505 must be decrypted, semantic digest reader 505 will do that as well. In some cases, that may require sending the encrypted semantic digest to a remote location that has the proper key.

Semantics reader 507 reads semantic information 305. If analog form 503 is a document, semantics reader 507 is a scanner which provides its output to OCR software. With other images, the scanner provides its output to whatever image analysis software is required to analyze the features of the image that make up semantic information 305. If analog form 503 is audio, the audio will be input to audio analysis software. Once the semantics information has been reduced to semantics data 509, it is provided to semantic digest maker 511, which makes a new semantic digest 513 out of the information. To do so, it uses the same technique that was used to make old semantic digest 515. Comparator 517 then compares old semantic digest 515 with new semantic digest 513; if the digests match, comparison result 519 indicates that analog form 203 is authentic; if they do not, result 519 indicates that they are not authentic. What "match" means in this context will be explained in more detail below.

"Matching" semantic digests

With the digests that are normally used to authenticate digital representations, exact matches between the old and new digests are required. One reason for this is that in most digital contexts, "approximately correct" data is useless; another is that the one-way hashes normally used for digests are "cryptographic", that is, the value of the digest reveals nothing about the

BNSDOCID: <WO___9957623A2_I_>

value from which it was made by the hash function, or in more practical terms, a change of a single bit in the digital representation may result in a large change in the value produced by the hash function. Since that is the case, the only comparison that can be made between digests is one of equality.

In the context of authenticating analog forms, the requirement that digests be equal causes difficulties. The reason for this is that reading semantic information from an analog form is an error-prone operation. For example, after many years of effort, OCR technology has gotten to the point where it can in general recognize characters with 98% accuracy when it begins with a clean copy of a document that is simply formatted and uses a reasonable type font. Such an error rate is perfectly adequate for many purposes; but for semantic information of any size, a new digest will almost never be equal to the old digest when the new digest is made from semantics data that is 98% the same as the semantics data that was used to make the old semantic digest. On the other hand, if the semantics data obtained from the analog form is 98% the same as the semantics data obtained from the digital representation, there is a very high probability that the analog form is in fact an authentic copy of the digital representation.

Precise matches

Of course, if the semantic information is limited in size and tightly constrained, it may be possible to require that the digests be exactly equal. For example, many errors can be eliminated if what is being read is specific fields, for example in a check or identification card, and the OCR equipment is programmed to take the nature of the field's contents into account. For example, if a field contains only numeric characters, the OCR equipment can be programmed to treat the letters o and O as the number 0 and the letters l,i, or I as the number 1. Moreover, if a match fails and the semantic information contains a character that is easily confused by the OCR equipment, the character may be replaced by one of the characters with which it is confused, the digest may be recomputed, and the match may again be attempted with the recomputed digest.

BNSDOCID: <WO 9957623A2 I >

Fuzzy matches

Where the semantic information is not tightly constrained, the digests must be made in such a fashion that closely-similar semantic information produces closely-similar digests. When that is the case, matching becomes a matter of determining whether the difference between the digests is within a threshold value, not of determining whether they are equal. A paper by Marc Schneider and Shih-Fu Chang, "A Robust Content Based Digital Signature for Image Authentication", in: *Proceedings of the 1996 International Conference on Image Processing*, presents some techniques for dealing with related difficulties in the area of digital imaging. There, the problems are not caused by loss of information when a digital representation is used to make an analog form and by mistakes made in reading analog forms, but rather by "lossy" compression of images, that is, compression using techniques which result in the loss of information. Because the lost information is missing from the compressed digital representation, a digest made using cryptographic techniques from the compressed digital representation will not be equal to one made from the digital representation prior to compression, even though the compressed and uncompressed representations contain the same semantic information. Speaking generally, the techniques presented in the Schneider paper deal with this problem by calculating the digest value from characteristics of the image that are not affected by compression, such as the spatial location of its features. Where there are sequences of images, the digest value is calculated using the order of the images in the sequences.

Analogous approaches may be used to compute the semantic digest used to authenticate an analog form. For example, a semantic digest for a document can be computed like this:

1. Set the current length of a digest string that will hold the semantic digest to "0";
2. Starting with the first alphanumeric character in the document, perform the following steps until there are no more characters in the document:
   a. Select a next group of characters;
   b. For the selected group,

BNSDOCID: <WO___9957623A2_I_>

i. replace characters in the group such as O,0,o; I,i,l,1; or c,e that cause large numbers of OCR errors with a "don't care" character;

ii. make a hash value from the characters in the group;

iii. append the hash value to the semantic digest string;

c. return to step (a).

3. When there are no more characters in the document, make the semantic digest from the digest string.

When computed in this fashion, the sequence of values in the semantic digest string reflects the order of the characters in each of the sequences used to compute the digest. If the sequence of values in the new semantic digest that is computed from the analog form has a high percentage of matches with the sequence of values in the old semantic digest, there is a high probability that the documents contain the same semantic information.

Applications of authentication with analog forms

One area of application is authenticating written documents generally. To the extent that the document is of any length and the digest is computed from a significant amount of the contents, the digest will have to be computed in a fashion which allows fuzzy matching. If the digest is computed from closely-constrained fields of the document, exact matching may be employed.

Another area of application is authenticating financial documents such as electronic cash, electronic checks, and bank cards. Here, the fields from which the digest is computed are tightly constrained and an exact match may be required for security. In all of these applications, the digest or even the semantic information itself would be encrypted as described above to produce a digital signature.

Universal Paper & Digital Cash

BNSDOCID: <WO___9957623A2_I_>

Digital cash is at present a purely electronic medium of payment. A given item of digital cash consists of a unique serial number and a digital signature. Authentication using semantic information permits digital cash to be printed as digital paper cash. The paper cash is printed from an electronic image which has a background image, a serial number, and a money amount. The serial number and the money amount are the semantic information. The serial number and the money amount are used to make a digital signature and the digital signature is embedded as an electronic watermark into the background image. The paper cash can be printed by any machine which needs to dispense money. Thus, an ATM, can dispense digital paper cash instead of paper money. Similarly, a vending machine can make change with digital paper cash and a merchant can do the same. The digital paper cash can be used in the same way as paper money. When a merchant (or a vending machine) receives the digital paper cash in payment, he or she uses a special scanner (including OCR technology and a watermark reader) to detect the watermark (i.e. the serial number and money amount) from the printed image, and send them to the bank for verification in the same fashion as is presently done with credit cards.

Digital Checks

Digital checks can be made using the same techniques as are used for digital paper cash. The digital check includes a background image, an identifier for the bank account, an amount to be paid, and the name of the payer. The payer's private key is used to make a digital signature from at least the identification of the bank and the amount to be paid, and the digital signature is embedded as an electronic watermark in the background image. Writing a digital check is a three-step process: enter the amount, produce the digital signature from the bank account number and the amount using the payer's private key, and embed the digital signature into the background image. The bank verifies the check by detecting the watermark from the digital check., decrypting the digital signature with the payer's public key, and comparing the bank account number and the amount from the image with the bank account number and the amount on the face of the check. A digital check can be used in either electronic form or paper form. In the latter case, a scanner (including OCR technology and watermark reader)

BNSDOCID: <WO___9957623A2_I_>

is needed to read the watermark from the paper check.

Authentication of identification cards

The techniques described above for authenticating digital paper cash or digital checks can be used with identification cards, including bankcards. The card number or other identification information appears on the face of the card, is encrypted into a digital signature, and is embedded as a digital watermark in the background image of the bankcard. The encryption can be done with the private key of the institution that issues the card. The merchant uses use a scanner to detect the digital signature (i.e. card number or other ID) from the card, and compare the signature with the authentication stored inside the card. This technique can of course be combined with conventional authentication techniques such as the holographic logo.

Active watermarks: FIGs. 6-8

Heretofore, digital watermarks have been nothing more than labels. They have typically contained information such as identifiers for the owner and creator of the digital representation and access control information, for example, whether the digital representation may be copied or changed. Any kind of information can, however be placed in a digital watermark. If the information in the watermark describes an action to be taken, the watermark becomes *active*, and the digital representation that contains the active watermark becomes active as well. This is the reverse of the usual practice of encapsulating a digital representation in a program, as is done for example with Microsoft Active Documents. Since digital watermarks are used in digital systems, the simplest way to make a watermark active is to include program code in it which can be executed by the computer system upon which the digital representation is currently resident. From the point of view of function, the code may be in any language for which the computer system can execute code. Practically, however, the code is best written in a language such as Java™ or Perl for which most modern computer systems have interpreters.

BNSDOCID: <WO 9957623A2 I >

FIG. 6 is an overview of a system 601 for making an active watermark 619. The watermark is made from watermark information 603, which contains owner information 605, access information 607, and owner-defined information 609 as before, but additionally contains code 611. Code 611 may be standard code for a given class of digital representations, or it may be defined specifically for a given digital representation. Code 611 may of course also use the other information in watermark information 603 as data. Watermark information 603 and digital representation 613 are input into watermark maker 615, which outputs digital representation 617, which is digital representation 613 modified to include watermark 619 made from watermark information 603. Since watermark information 603 includes code 611, watermark 619 is an active watermark.

FIG. 11 shows a preferred embodiment of access information 607. It includes fields as follows:

- an 8-bit permission (P) field which indicates the kind of access the user may have: among the kinds are access which permits display, access which permits storing a local copy, and access which permits printing.
- a four-bit sensitivity field whose value indicates the degree of sensitivity of the contents of the digital representation;
- a 32-bit allowed location field which contains the IP address at which the digital representation is permitted to be located;
- A 32-bit allowed period field which contains a period of time for which use of the digital representation has been permitted; and

FIG. 7 is an example of a program which might be found in code 611. Program 701 is written in the Java programming language. It is then compiled into Java bytecodes which are interpreted by a Java interpreter. These bytecodes are included in the digital watermark. When program 701 is executed, a message indicating that digital representation 617 containing the active watermark has been displayed is sent via the Internet to a system that has been set up to monitor the display of digital representation 617, perhaps for the purpose

BNSDOCID: <WO___9957623A2_I_>

of computing license fees. Line 703 of the code sets up a socket s by means of which a datagram may be sent to the monitoring system. Line 709 of the code finds the current Internet address a of the monitoring system which is specified at 705 by the name ▪syscop.crg.edu▪. Line 1715 makes a new datagram packet for the message; it includes the message content, ▪XYZ Displayed▪ and the internet address a. Line 1719, finally, uses the send operation associated with the socket s to send the message, which the Internet will deliver to the destination specified by a.

FIG. 8 shows a system 801 for executing the code in active watermark 619. Digital representation 617 containing active watermark 619 is input to watermark reader 803, which extracts watermark information 603 from active watermark 619. Info 603 includes code 611, which watermark reader 803 provides to code interpreter 805. Code interpreter 805 interprets code 611 to provide instructions which are executable by the computer system upon which code interpreter 805 is running. In some embodiments, code interpreter is an interpreter provided by the computer system for a standard language such as Java; in others, interpreter 805 may be provided as a component of watermark reader 803. In such embodiments, code 611 may be written in a language specifically designed for active watermarks.

An active watermark 619 can cause the computer system in which the active watermark is read to perform any action which can be described by the code contained in the active watermark. The only limitations are those imposed by the fact that the code is part of a watermark. One of these limitations is code size: code contained in a watermark must necessarily be relatively short; this limitation can be alleviated by compressing the code using a "non-lossy" compression technique, that is, one which does not result in the loss of information. Another of the limitations is that damage to the watermark may result in damage to the code; consequently, active watermarks may not work well in situations where the digital representation 617 is involved in "lossy" manipulations, i.e., manipulations that cause loss of information in digital representation 617. Examples of such lossy manipulations

BNSDOCID: <WO___9957623A2_I_>

are editing the digital representation, lossy translation of the digital representation from one format into another, lossy compression of the digital representation, and producing a new digital representation from an analog form made from an old digital representation (as would be the case, for example, if the code were obtained by reading the watermark from a paper copy of a document).

Of course, techniques like those discussed above with reference to exact matches of digests can be applied to recover code from a damaged watermark or from an analog form, and to the extent that such techniques are successful, active watermarks can be used even where lossy manipulations have taken place. For example, the watermark of an analog form may contain not only authentication information, but also code. If a copy machine contained a watermark reader and an interpreter for the code used in the active watermark, the active watermark could be used, for instance, to prevent the copy machine from copying the analog form.

Among the things that can be done with active watermarks are the following:

- Customizing the manner in which the digital representation containing the watermark is treated. Code 611 may differ for classes of digital representations, or may even be particular to a single digital representation;
- Having a digital representation send a message whenever it is displayed, copied, printed, or edited; for example, whenever a document with an active watermark stored on a Web server is downloaded from the server, the active watermark can cause a message containing billing information to be sent to a billing server.
- Having the digital representation obtain locally-available information, which will then govern the behavior and usage of the digital representation;
- Having a digital representation take protective action when a user tries to do something with it that is not permitted by access information 603; the protective action can range from a warning through sending a message or blocking the intended action to destroying the digital representation that contains the watermark.

BNSDOCID: <WO___9957623A2_I_>

Watermark agents

Digital representations pose special problems for their owners because, like all digital data, they can be easily copied and distributed across a network. These properties of digital data, however, also make it possible to automate monitoring of the distribution and use of watermarked digital representations. One way to do this is the watermark spider. As mentioned in the *Description of the Prior Art*, the watermark spider follows URLs to Web pages, which it retrieves and inspects for watermarks. If it finds one that is of interest, it reports its findings to a monitoring program. There are two problems with the watermark spider: the first is that it is limited to digital representations which are accessible by URLs that are available to the public. Thus, the watermark spider would not be able to locate a copy of a digital representation on a WWW client, as opposed to on a WWW server. The other problem is that the spider must fetch each digital representation to be examined across the network. Since digital representations are often large, the need to do this adds substantially to the volume of network traffic.

Both of these problems can be solved by means of a *network watermark agent*, that is, a watermark monitor which uses the network to move from system to system where digital representations of interest might be stored. At each system, the watermark agent examines the system's file system for digital representations that have watermarks of interest. If the watermark agent finds such a watermark, it may send a message with its findings via the network to a monitoring program. The watermark agent is thus able to monitor digital representations that are not available via public URLs and uses network bandwidth only relatively rarely and only to send messages that are small in comparison with digital representations. In the following, the creation of a watermark agent and its behavior in a system will both be explained in detail.

Creating a watermark agent: FIG. 9

FIG. 9 shows a watermark monitoring system 901 which creates and dispatches a watermark

BNSDOCID: <WO___9957623A2_I_>

agent 925 across a network 103 and responds to messages from the watermark agent Watermark agent 925 is a program which is able to send itself from one node to another in network 103. In each node, it searches for watermarked documents and sends messages 935 containing its findings to monitoring system 901, where message handler 920 deals with the message, often by adding information to management data base 903.

Continuing in more detail, agent 925 has two main parts: agent code 927, which is executed when agent 925 reaches a node, and agent data 929, which contains information used by agent 925 in executing the code and in moving to the next node. At a minimum, agent code 927 will include code which searches the node for files that may contain watermarks, code that makes and sends any necessary messages to monitoring system 901, code that clones agent 925, and code that sends the clone on to the next node. As with the code in active watermarks, code 927 may be written in any language which can be executed in a node; either standard languages such as Java or a specialized watermark agent language may be used.

FIG. 12 provides an example written in the Java language of code 1201 that a watermarking agent 925 might execute. Code 1201 searches the file system of the network node at which agent 925 is presently located for images files, checks each image file for a watermark, if it finds a watermark, it performs the action required by the watermark and the node, and makes a message containing a list of the actions it performed.

Continuing in more detail, code 1201 has two main sections, initialization 1203 and checking loop 1213. In initialization 1203, the first step is to instantiate a file filter to filter the files in the node's file system (1205). Then a function of the filter which locates image files is used to make a list `filenames` of the names of the image files in the file system (1207). Thereupon, information about the environment of the node that the agent needs to check watermarks is retrieved and placed in a variable `env` (1209); finally, a data structure called `results` is created to hold the results of the watermark checks 1211.

BNSDOCID: <WO___9957623A2_I_>

In loop 1213, each file in `filenames` is examined in turn for a watermark (1215); if one is found, the actions indicated at 1217 are performed; first, the contents of the watermark are compared with the environment information to obtain a result called `match` (1219). Then `match` is passed to a function which takes an action as determined by the value of `match` and returns a value `result` which represents the result of the action (1221); finally, `result` is added to the data structure `results` (1223). then, at 1225, `results` is returned. Depending on how the watermark agent is being used, `results` can then be sent in a message to monitoring system 901.

Continuing in more detail with agent data 929, agent data 929 includes a map 931, digital representation description 933, keys 934, and parameters 921. Map 931 is a list of addresses in network 103. Each address specifies an entity in network 103 that can provide an environment in which agent 925 can operate. The address may for example be an E-mail address or an IP address. Digital representation description 933 may be any information that describes the digital representations the agent is looking for. There may be a filter for the file names and there may also be identification information from the watermark. For example, if the files to be examined are `.bmp` files, the filter might specify `*.bmp`, indicating that all files with the `.bmp` suffix are to be examined. If a watermark key is needed to read the watermark, keys 934 will contain that key and if the messages sent to monitor system 901 are to be encrypted, keys 934 will contain the key to be used in encrypting the messages. Any available technique may be used to keep the keys secure. In a preferred embodiment, the parameters include

- the email address for messages sent by the agent;
- whether to report on files to which agent 925 had no access;
- date of last monitoring and whether to check only files updated since that date;
- whether to execute code 611 in an active watermark 619; and
- termination conditions for agent 925.

Agent 925 is produced by agent generator 923, which can be implemented as a component

BNSDOCID: <WO___9957623A2_I_>

of digital representation manager 131. Agent generator 923 makes agent 925 from information in management data base 903 and agent parameters 921, which here are shown being provided interactively by a user of monitoring system 901, but may also be stored in management data base 903. The information in management data base 903 includes agent template 905(i), which is one of a number of templates that are used together with parameters 921 and other information in management data base 903 to generate agent code 927 for different kinds of agents 925. Suspicious sites 907 is a list of network locations which might be worth examining. One source of information for sites that should be on the list of suspicious sites 907 is of course messages received from previously-dispatched agents. Network information 909 is information about the network. Suspicious sites 907 and network information 909 are used together to make map 931 in agent 925. Digital representation information 911, finally, contains information about the digital representations that the agent will be looking for. The information is used to make DR Description 933. Information 911(i) for a given digital representation or group of digital representations may include a watermark key 913 for the digital representation's watermark and information from the watermark including owner ID 915, user ID 917, and permitted use information 919. User ID 917 is an identification for the user to whom the digital representation was downloaded. Once agent 905 has thus been created by monitor system 901, agent 925 clones itself, makes the clone into the kind of message required for the first entity specified in map 931, and sends the message to the first entity. Thereupon, agent 925 terminates itself.

Watermark agents in network nodes: FIG. 10

FIG. 10 shows those components of a network node 1001 which are involved in the monitoring of the node by a watermark agent 925 The components include:

- agent engine 1003, which provides the environment in which agent 925 executes its code and which is the entity to which the message containing agent 925 is addressed;
- file storage 1031, which contains the digital representations 1023 that are of interest to agent 925;
- file system 1029, which makes the digital representations 1023 accessible as files;

BNSDOCID: <WO___9957623A2_I_>

- watermark reader 1019, which reads the watermarks; and
- code interpreter 1011, which interprets code in agent 925 and may also interpret the code in active watermarks, if that code is written in the same language as the code used in agent 925.

SC 1035 is an optional secure coprocessor whose functions will be explained in more detail in the discussion of security.

Operation of components 1001 is as follows: When the message containing agent 925 arrives in agent engine 1003 from network 103, agent engine 1003 extracts agent 925 from the message and, at a convenient time, uses code interpreter 1011 to begin executing its code. What the code does is of course arbitrary. Typically, it will do the following:

1. Send a message to system 901 indicating its arrival in the node;
2. Obtain the file filter from DRDESC 993 and give it to spider 1009 to make a list of files that match the filter;
3. For each file on the list, do the following:
   a. use spider 1009 to get the file ID for the file;
   b. give file ID1021 to watermark reader 1019, which uses the watermark key from keys 934 to read the watermark, if any;
   c. receive the watermark content 1017;
   d. process watermark content 1017 as specified in code 927. Actions might include sending a message to system 901 or passing the code and data 1015 from an active watermark to code interpreter 1011 for execution and receiving data 1013 in return;
4. When all of the files have been processed,
   a. sending a message to monitor system 901 with summary information about the results of the visit and the next node to be visited;
   b. making a clone of agent 925 and sending the clone to the next address specified in map 931; and
   c. terminating agent 925.

BNSDOCID: <WO___9957623A2_I_>

As previously indicated, what a watermark agent can do is essentially arbitrary If the documents being dealt with by the watermark agent have active watermarks, there are any number of ways of dividing the work of processing of documents of interest between the code in the watermark agent and the code in the active watermark. For instance, in the example above, step 3(d) above could consist simply of executing the code in the document's active watermark.

The actions performed in step3(d) will typically be performed when the information in the watermark does not match the time or place where agent 925 found the file or the time and/or place are inappropriate for the file's access privileges. The action may be one of a pre-defined set specified by parameters in parameters 921, it may be one defined by agent 925's code 927, or it may be one defined by an active watermark. Among the pre-defined actions are:

1. Destroy the file if the file's sensitivity level is very high;
2. Remove the file to a safe place if the sensitivity level is medium;
3. If the sensitivity level is low,
    a. Warn the local administrator or webmaster of the violation if the sensitivity is low;
    b. Warn the recipient of the violation if the sensitivity is low; or
    c. Send a message to the file's owner reporting the violation if the sensitivity is low;
4. If the sensitivity level is very low, send a message to monitor 901 without disturbing the local host and local administrator .

Before going on to the next destination, watermark agent 925 may, wait for a message from monitor 901 containing information about the next destination; the information may include:

- The time of the last visit by an agent to the destination;
- Information about the destination, for example detailed information about the digital representations to be examined there.

BNSDOCID: <WO___9957623A2_I_>

Nontraveling watermark agents

An important difference between a watermark agent and a watermark spider is that the watermark agent interacts with the document in the system where the document is being stored or processed, and can thus perform far more functions than a watermark spider can. A further consequence of this difference is that a watermark agent need not travel, but can simply be incorporated as a permanent component of a system. For example, a copier could include a watermark agent that read the watermarks of paper documents being copied and prevented the copier from copying a document when its watermark indicated that the document was not to be copied. An important application of such a non-traveling watermark agent would be to prevent the copying of paper digital cash.

Of course, if the copier had access to a network, even the "non-traveling" watermark agent could at least travel via the network to the copier, and the network would provide a convenient way of updating the copier's watermark agent. "Non-traveling" watermark agents could of course be distributed in a similar fashion to any system accessible via the network.

Security considerations

In some cases, for example in private military or business networks or systems, agent 925 may not operate in a hostile environment, and monitor 901 and agent engine 1003 may even be implemented as integral parts of the operating system. In most cases, however, agent 925 will be operating in an environment which is hostile in at least four respects:

- The node to which agent 925 sends itself is properly suspicious of messages from outside that contain code to be executed on the node;
- to the extent that users on the node have violated the conditions under which they received a digital representation, they will want to hide their behavior and/or disable agent 925;
- users on the node may want access to the keys and other data carried by agent 925;

BNSDOCID: <WO 9957623A2 I >

and

- other users of network 103 may be interested in the content of the messages being exchanged between agent 925 and monitor 901.

The first of these problems is the "malicious agent problem". It is general to systems that download and execute code, and the same solutions that are used in those cases can be applied to agent engine 1003 and agent 925. For example, if the watermark agent's code is written in Java, the system on which it is run will have whatever protections are provided by the Java interpreter. If managers of nodes are reasonably certain that agent engine 1003 and agents 925 will not do any damage to the nodes, they can be made to accept engine 1003 and agents 925 simply as a condition of downloading digital representations. For example, the transaction by which a digital representation manager downloads a digital representation to a node might include a message to agent engine 1003 confirming the existence and operability of agent engine 1003. If the message were not properly answered, the digital representation manager might require that the node download and install agent engine 1003 before proceeding further with the transaction.

The remainder of these problems are termed "malicious node problems". They can be solved by standard cryptographic techniques, as described in Schneier, *supra*. For example, the digital representation manager and each agent engine 1003 might have a public key-private key pair. In that case, network information 909 would include the public key for the agent engine 1003 at a given node and the public keys for the agent engines 1003 in the nodes to be visited would be included in map 931. Any message sent by the digital representation manager or by an agent 925 to an agent engine 1003 can be encrypted using agent engine 1003's public key and any message sent by an agent engine 1003 or an agent 925 to a digital representation manager can be sent using the digital representation manager's public key. The public key for the digital representation manager can of course be included in agent 925's keys 934. Authentication of messages can be done using standard digital signature techniques; for example, agent data 929 might include a digital signature from the digital

BNSDOCID: <WO___9957623A2_I_>

representation manager for agent 925, messages from the digital representation manager to agent engine 1003 can include the digital representation manager's digital signature, and messages from agent engine 1003 can include agent engine 1003's digital signature.

If the watermarks are made using encryption techniques, as described in E. Koch and J. Zhao, "Towards Robust and Hidden Image Copyright Labeling", *supra*, the agent must have a way of decrypting the watermark. Depending on the situation, the watermark may be encrypted with the watermark agent's public key and authenticated with a digital signature in the same fashion as other messages to the agent engine or the watermark may have its own key 913. In the former case, the, the watermark agent's private key must be protected and in the latter, watermark key 913 must be protected, since access to the key would permit those intent on stealing digital representations to remove or alter the digital representation's watermark. While agent 925 is in transit, watermark key 913 can be protected by encryption in the same fashion as the rest of the information in agent 925; once agent 925 has been decrypted, watermark key 913 and agent engine 1003's private key must be protected in the node currently being visited by agent 925. Agent engine 1003's private key must further be protected to prevent a user of the node currently being visited by agent 925 from using the private key to decrypt messages addressed to agent engine 1003 or append agent 1003's digital signature.

One way of solving these key protection problems is a secure coprocessor, as described in J.D. Tyger and Bennet Lee, *Secure Coprocessors in Electronic Commerce Applications*, FIRST USENIX WORKSHOP ON ELECTRONIC COMMERCE, JULY 1995. As shown at 1033, a secure coprocessor includes secure storage 1035 and a secure processor 1045. Secure storage 1035 may only be accessed via secure processor 1045, and secure coprocessor 1033 is built in such a fashion that any attempt to access the information in secure coprocessor 1033 other than via secure processor 1045 results in the destruction of the information. Secure coprocessor 1033 is able to write information to and read information from secure storage 1035 and also does encryption and decryption and makes and verifies

BNSDOCID: <WO___9957623A2_I_>

digital signatures. These operations may be done entirely by executing code stored in secure storage 1035 or by means of a combination of code and specialized hardware devices, as shown at 1047 and 1049. The keys used in encryption, decryption, and in making digital signatures and verifying them are stored in secure storage 1035. Shown in FIG. 10 are WMkey 913 for the watermark, monitor public key 1039, agent engine public key 1041, and agent engine private key 1043. In the case of the public keys, storage in secure storage 1035 is simply a matter of convenience, and secure processor 1045 may provide access to the public keys in response to requests from components of node 1001; in the case of WMkey 913 and agent engine 1003's private key 1043, the decrypted keys 913 and 1043 are used only within secure processor 1033.

In the context of system 1001, when a message encrypted with agent engine 1003's public key 1041 arrives in agent engine 1003, agent engine 1003 uses secure processor 1033 to decrypt the message; if the message contains an agent 925, agent engine 1003 also uses secure processor 1033 to verify that agent 925's digital signature is from the digital representation manager and to decrypt WMkey 913. The decrypted key is not returned to agent engine 1003, but is stored in secure storage 1035. SWM reader 1019 then uses secure coprocessor 1033 to decrypt the watermark in the digital representation currently being checked by agent 925.

Applications using watermark agents

Since a watermark agent is programmed, it can do literally anything. The flexibility of watermark agents is increased when their use is combined with that of active watermarks. One set of applications for watermark agents is monitoring the use of copyrighted digital representations for the copyright owner or a licensing agency. A copyright owner or licensing agency, for example, may use watermark agents to locate unlicensed copies of digital representations or to periodically monitor the use of licensed copies. A document with an active watermark could increment a usage count maintained in agent engine 1003 for a node each time it was printed and agent 925 could read the count on its visit to the node,

BNSDOCID: <WO___9957623A2_I_>

report the current count value back to management database 903, and reset the counter.

Another set of applications is monitoring the use of digital representations to avoid liability for infringement. For example, a corporation might want to be sure that it has no unauthorized digital representations in its network and that the authorized ones are being used in accordance with their license terms. The agent can monitor the use of the digital representations in the corporate network in the same fashion as it does for the licensing agency. In this instance, the monitoring might even include destroying illegal copies.

Yet another set of applications is preventing unauthorized copying, scanning, or printing. This can be done by means of "nontraveling" watermark agents on servers and clients in the network or even by means of "nontraveling" watermark agents built into devices such as copiers, scanners, or printers. For example, if a "No copy" watermark is embedded in currency and a photocopier has an agent that looks for such a watermark and inhibits copying when it finds the watermark, the photocopier will not make copies of currency.

Watermark agents can also be used to enforce military or corporate document security rules. In such an application, the document's security classification would be embedded in it as a watermark and the watermark agent would search the military or corporate file systems and networks for documents that were not being dealt with as required by their security classification. Examples would be documents that were in the wrong place or had been kept longer than a predetermined period. Actions taken by the agent can range from reports and warnings through changing the access rights to the document or moving the document to a safe location to immediate destruction of the out-of-place document. Again, the agent that does this need not travel, but may simply be a permanent component of the file system.

Watermark agents, finally, can be used to find lost documents in military or business file systems or networks. If each document has a unique identifier associated with it and that identifier is on the one hand kept in a database and on the other hand incorporated into a

BNSDOCID: <WO 9957623A2 I >

watermark in the document, a watermark agent can simply be given the universal identifier and sent to search the file system or network for the document. Once the agent has found it, it can report its location to whomever sent the agent out.

Conclusion

The foregoing *Detailed Description* has disclosed to those skilled in the relevant arts how to make and use documents with authentication that withstands conversion between an analog form and a digital representation of the document, how to make and use digital representations with active watermarks, and how to make and use watermark agents, including mobile watermark agents and has further disclosed the best mode presently known to the inventors for making such authentications, making active watermarks, and making watermark agents. The disclosed techniques are exceedingly general and may be implemented in many different ways for many different purposes. For example, the authentication techniques may be based on any kind of semantic information and there are many ways of deriving the authentication information from the semantic information, placing the authentication information in the digital representation or the analog form, and comparing the authentication information. Similarly, the program code for an active watermark may be written in any programming language, may be in source or object form, and may, when executed, perform arbitrary operations. Watermark agents, too, may perform arbitrary actions and employ various techniques for sending messages and traveling from node to node in a network. The watermark agents can of course perform authentication information and can execute code from active watermarks.

Since the techniques are so general and may be implemented in any number of ways, the *Detailed Description* is to be regarded as being in all respects exemplary and not restrictive, and the breadth of the invention disclosed herein is to be determined not from the *Detailed Description*, but rather from the claims as interpreted with the full breadth permitted by the patent laws.

BNSDOCID: <WO___9957623A2_I_>

What is claimed is:

1. Improved digital watermarking apparatus of the type that receives information and makes a digital watermark in a digital representation that contains the information, the improvement comprising:

including program code in the received information,

whereby an apparatus that reads the digital watermark may cause the program code to be executed.

2. The improved digital watermarking apparatus set forth in claim 1 wherein:

the digital watermark has no substantial effect on programs that process the digital representation,

whereby no modification of such programs is required.

3. The improved digital watermarking apparatus set forth in claim 1 wherein:

the digital watermark is encrypted.

4. The improved digital watermarking apparatus set forth in claim 3 wherein:

the digital watermark is associated with authentication information.

5. The improved digital watermarking apparatus set forth in claim 1 wherein:

the program code specifies an action to be performed if an operation is performed on the digital representation.

6. The improved digital watermarking apparatus set forth in claim 5 wherein the operation comprises one or more of:

copying the digital representation, displaying or playing the digital representation, printing the digital representation, modifying the digital representation, modifying the digital representation's relationship to a file system, modifying access privileges associated with the digital representation, or further distributing the digital representation.

BNSDOCID: <WO___9957623A2_I_>

7. The improved digital watermarking apparatus set forth in any of claims 5 or 6 wherein the action comprises one or more of:

sending a message, inhibiting the action, or destroying the digital representation.

8. The improved digital watermarking apparatus set forth in claim 7 wherein the action of sending a message comprises one or more of:

sending a usage monitoring message to a usage monitoring system, sending a billing information message to a billing system, sending an informational message to a user for whom the operation is being performed, or sending a security message to a security system.

9. Improved digital watermark reading apparatus of the type which reads a digital watermark in a digital representation,

the improvement comprising:

a program code interpreter that causes program code included in the digital watermark to be executed.

10. The improved digital watermark reading apparatus set forth in claim 9 wherein:

the digital watermark has no substantial effect on programs that process the digital representation,

whereby no modification of such programs is required.

11. The improved digital watermark reading apparatus set forth in claim 9 wherein:

the digital watermark is encrypted.

12. The improved digital watermark reading apparatus set forth in claim 9 wherein:

the program code specifies an action to be performed if an operation is performed on the digital representation.

BNSDOCID: <WO___9957623A2_I_>

13. The improved digital watermark reading apparatus set forth in claim 12 wherein the operation comprises one or more of:

copying the digital representation, displaying or playing the digital representation, printing the digital representation, modifying the digital representation, modifying the digital representation's relationship to a file system, modifying access privileges associated with the digital representation, or further distributing the digital representation.

14. The improved digital watermark reading apparatus set forth in any of claims 12 or 13 wherein the action comprises one or more of:

sending a message, inhibiting the action, or destroying the digital representation.

15. The improved digital watermark reading apparatus set forth in claim 14 wherein the action of sending a message comprises one or more of:

sending a usage monitoring message to a usage monitoring system, sending a billing information message to a billing system, sending an informational message to a user for whom the operation is being performed, or sending a security message to a security system.

16. Improved apparatus for performing an operation on a digital representation, the apparatus having the improvement comprising:

a watermark reading apparatus that includes a program code interpreter for interpreting program code included in a digital watermark in the digital representation, the program code specifying an action to be performed if the operation is performed on the digital representation.

17. The improved apparatus set forth in claim 16 wherein:

the improved apparatus is a printing apparatus and the program code interpreter interprets the program code if the digital representation is printed thereby.

18. The improved apparatus set forth in claim 16 wherein:

BNSDOCID: <WO___9957623A2_I_>

the improved apparatus is a display apparatus and the program code interpreter interprets the program code if the digital representation is displayed thereby.

19. The improved apparatus set forth in claim 16 wherein:

the improved apparatus makes a copy of the digital representation and the program code interpreter interprets the program code if the digital representation is copied thereby.

20. The improved apparatus set forth in claim 16 wherein:

the improved apparatus distributes the digital representation via a network and the program code interpreter interprets the program code if the digital representation is distributed thereby.

21. The improved apparatus set forth in claim 16 wherein:

the improved apparatus is an editor for the digital representation and the program code interpreter interprets the program code if the digital representation is altered by the editor.

22. The improved apparatus set forth in claim 16 wherein:

the improved apparatus is a watermark agent and the program code interpreter interprets the program code if the watermark agent so indicates.

23. The watermark agent set forth in claim 22 wherein:

the digital representation resides in a node in a network; and

the watermark agent travels via the network to the node.

24. The improved apparatus set forth in any of claims 16 through 23 wherein:

the improved apparatus is implemented in a browser.

25. The improved apparatus set forth in claim 16 wherein:

BNSDOCID: <WO___9957623A2_I_>

the digital representation is made from an analog form that includes the digital watermark.

26. The improved apparatus set forth in claim 25 wherein:

the improved apparatus is implemented in a copier that makes a copy of the analog form.

27. A method of making a digital watermark comprising the steps of:

receiving program code to be included in the digital watermark; and

making the digital watermark with the program code.

28. A method of reading a digital watermark comprising the steps of:

reading the digital watermark; and

if the digital watermark contains program code, executing the program code.

29. An improved digital watermark,

the improvement comprising:

information in the digital watermark that includes program code.

30. A method of receiving information about a digital representation that includes a watermark that contains program code which is executed when an operation is performed on the digital representation and if executed, causes a message to be sent,

the method comprising the steps of:

distributing the digital representation; and

receiving the message.

BNSDOCID: <WO___9957623A2_I_>

31. The method set forth in claim 30 further comprising the step of:

adding the watermark to the digital representation.

32. A method of obtaining information about digital representations that contain watermarks, the method being employed in a network and

comprising the steps of:

sending an agent that includes a program that reads watermarks via the network to a system on which digital representations with watermarks may be resident; and

receiving a message from the agent program via the network indicating results of the agent program's reading of watermarks in the system.

33. The method set forth in claim 32 wherein:

the watermark includes program code; and

the agent program causes the watermark's program code to be executed in the system.

34. The method set forth in claim 32 wherein:

the agent program further sends the agent to a further system via the network.

35. The method set forth in claim 34 wherein:

the agent further includes a map of nodes in the network to be visited by the agent; and

the agent program selects the further system as specified in the map.

36. The method set forth in claim 32 wherein:

the agent further includes a description of the digital representations whose watermarks are to be read by the agent.

37. The method set forth in claim 32 wherein:

the agent further includes a first key used to encrypt the message.

BNSDOCID: <WO___9957623A2_I_>

38. The method set forth in claim 32 wherein:

the watermark is encrypted; and

the agent further includes a second key used to decrypt the watermark.

39. The method set forth in claim 32 wherein:

the system includes a watermark engine to which the agent is sent; and

the method further comprises the step of :

encrypting the agent so that it can only be decrypted by the watermark engine.

40. The method set forth in any of claims 32 through 39 wherein:

the message indicates a violation of security policy with regard to a digital representation resident on the system

whereby the agent is used to enforce security policy.

41. The method set forth in any of claims 32 through 39 wherein:

the message indicates that a digital representation resident on the system at least appears to be an unauthorized copy.

42. The method set forth in claim 41 wherein:

the message indicates that the copy at least appears to be an unauthorized copy of a copyrighted work,

whereby the agent is used to enforce an owner's copyright rights in the copy.

43. The method set forth in claim 41 wherein:

the message indicates that the copy at least appears to violate a security policy.

44. A method whereby a system which has received digital representations that contain watermarks may provide information about the received digital representations, the method

BNSDOCID: <WO 9957623A2 I >

being employed in a network and comprising the steps of:

receiving an agent in the system via the network, the agent including a program that reads watermarks and sends a message reporting results thereof via the network; and

executing the agent program in the system.

45. The method set forth in claim 44 wherein:

the agent program further sends the agent to a further system via the network.

46. The method set forth in claim 45 wherein:

the agent further includes a map of nodes in the network to be visited by the agent; and

the agent program selects the further system as specified in the map.

47. The method set forth in claim 44 wherein:

the agent further includes a description of the digital representations whose watermarks are to be read by the agent; and

the agent program selects the digital representations as specified in the description thereof.

48. The method set forth in claim 44 wherein:

the system includes a watermark engine to which the agent is sent; and

the method further comprises the step of :

encrypting the agent so that it can only be decrypted by the watermark engine.

49. A method of providing information about digital representations containing watermarks in a system which has access to a network and includes a watermark agent, the watermark agent including a program for reading a watermark and sending a message reporting a result thereof via the network,

the method comprising the steps of:

BNSDOCID: <WO___9957623A2_I_>

receiving a digital representation with a watermark in the system; and

executing the watermark agent's program.

50. The method set forth in any one of claims 44 through 48 and claim 49 wherein:

the watermark includes program code; and

the agent program causes the watermark's program code to be executed in the system.

51. The method set forth in any one of claims 44 through 48 and claim 49 wherein:

the agent further includes a first key used to encrypt the message.

52. The method set forth in any one of claims 44 through 48 and claim 49 wherein:

the watermark is encrypted; and

the agent further includes a second key used to decrypt the watermark.

53. The method set forth in any one of claims 44 through 48 and claim 49 wherein:

the agent's program further performs an operation on the digital representation containing the watermark.

54. The method set forth in claim 53 wherein:

the operation is one or more of changing the access rights to the digital representation, moving the digital representation, or deleting the digital representation.

55. The method set forth in any of claims 44 through 48 and claim 49 wherein:

the message indicates a violation of security policy with regard to a digital representation resident on the system

whereby the agent is used to enforce security policy.

56. The method set forth in any of claims 44 through 48 and claim 49 wherein:

the message indicates that a digital representation resident on the system at least

BNSDOCID: <WO___9957623A2_I_>

appears to be an unauthorized copy.

57. The method set forth in claim 56 wherein:

the message indicates that the copy at least appears to be an unauthorized copy of a copyrighted work,

whereby the agent is used to enforce an owner's copyright rights in the copy.

58. The method set forth in claim 56 wherein:

the message indicates that the copy at least appears to violate a security policy.

BNSDOCID: <WO___9957623A2_I_>

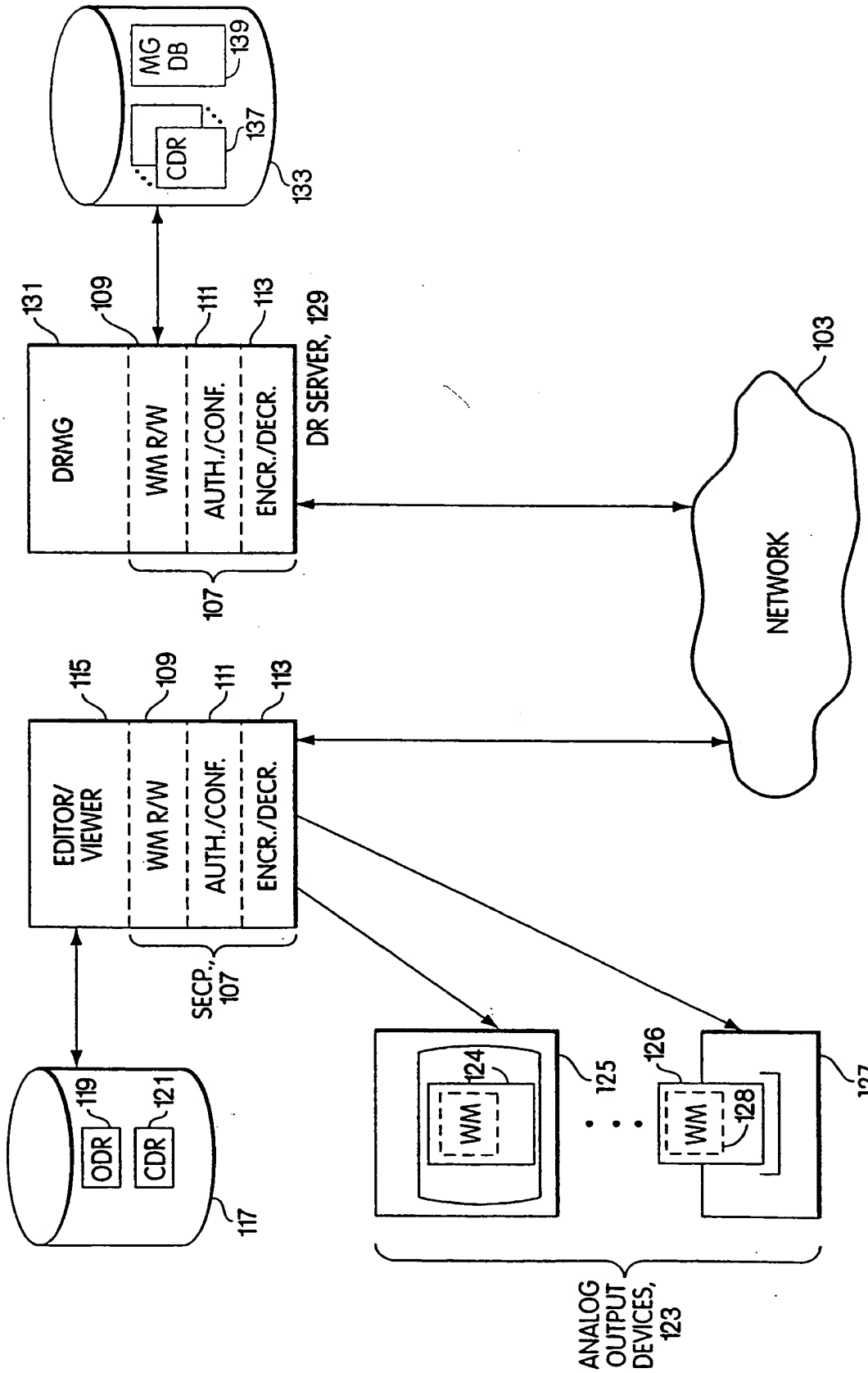


Fig. 1
Prior Art

BNSDOCID: <WO___9957623A2_I_>

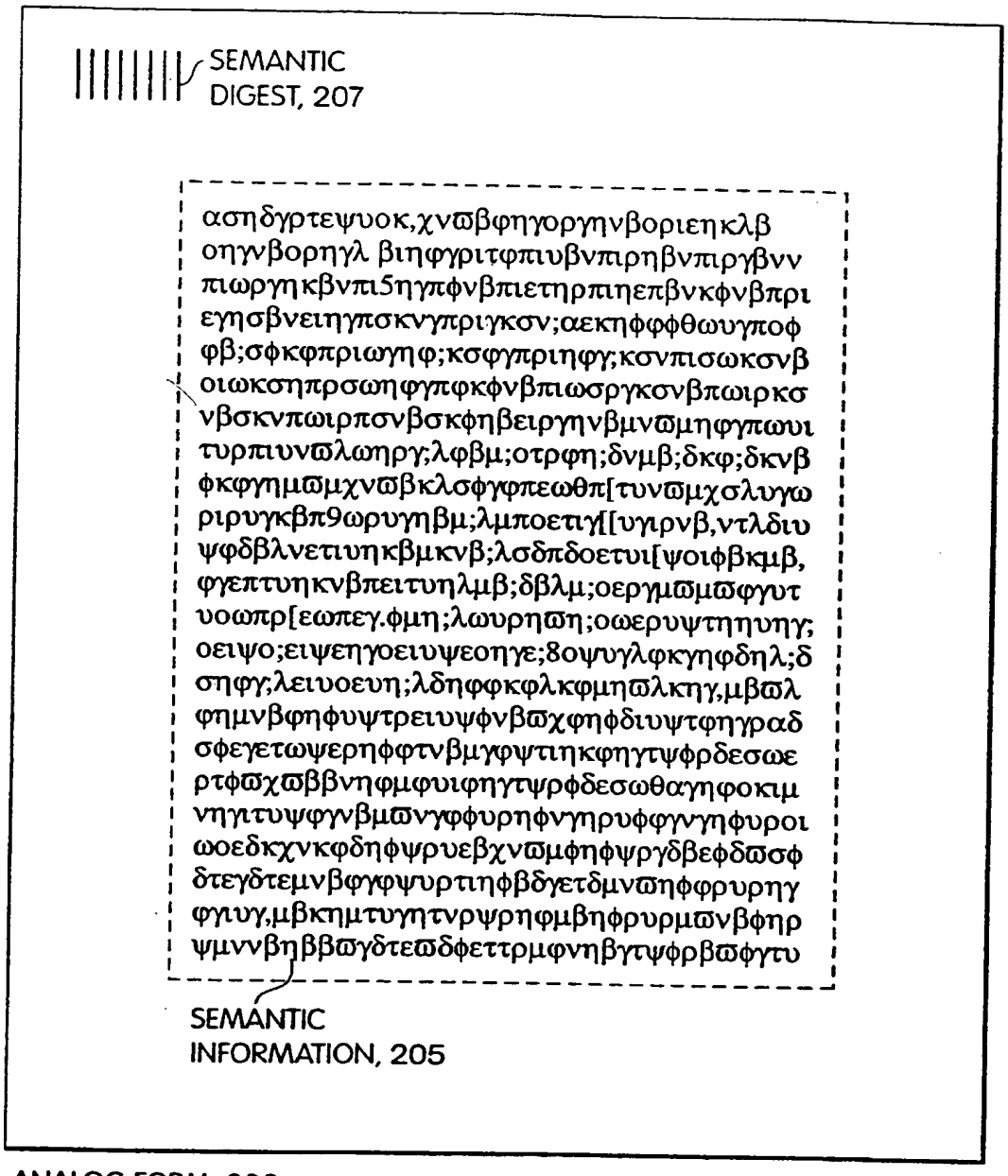


Fig. 2

BNSDOCID: <WO___9957623A2_I_>

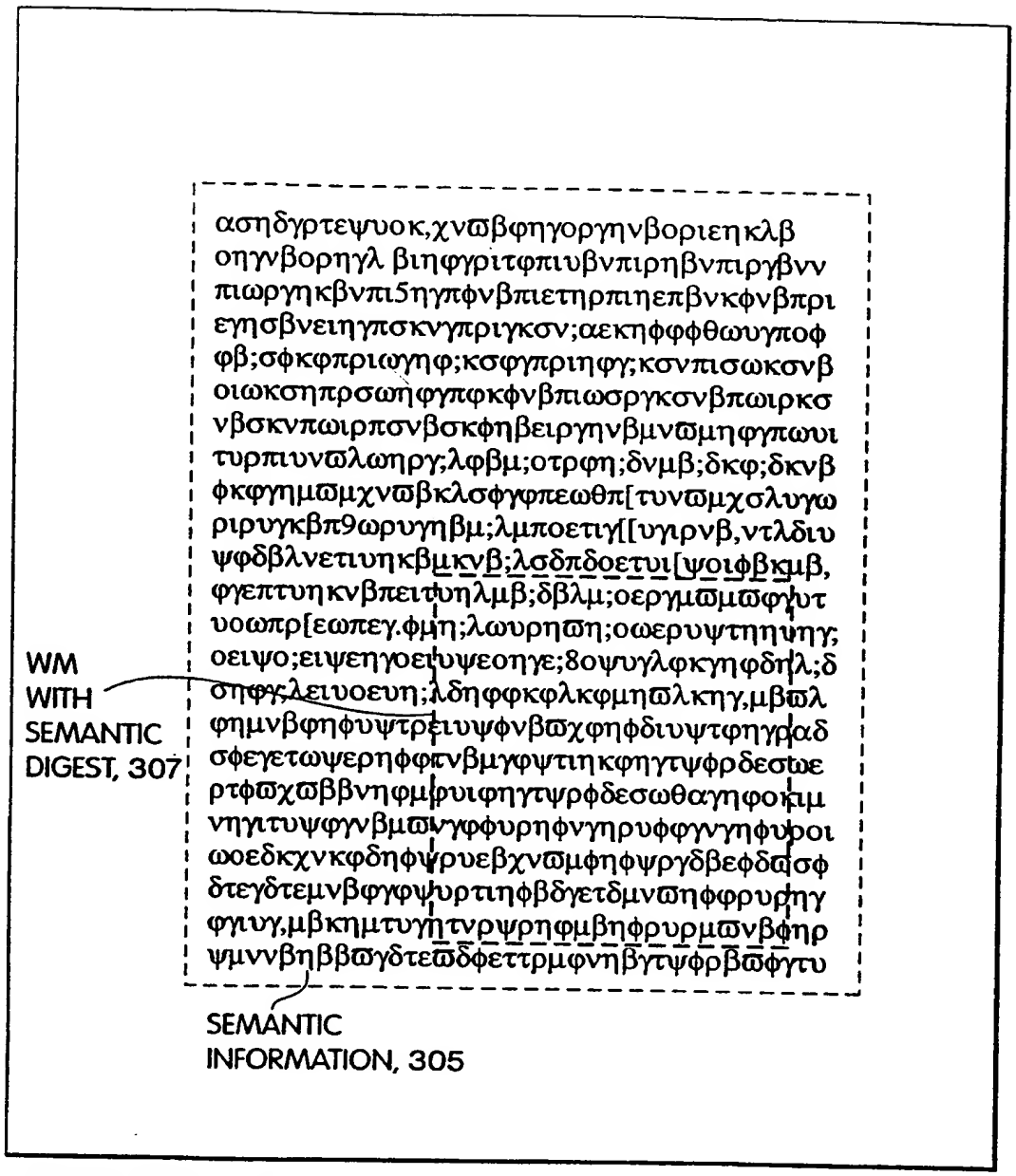


Fig. 3

BNSDOCID: <WO___9957623A2_I_>

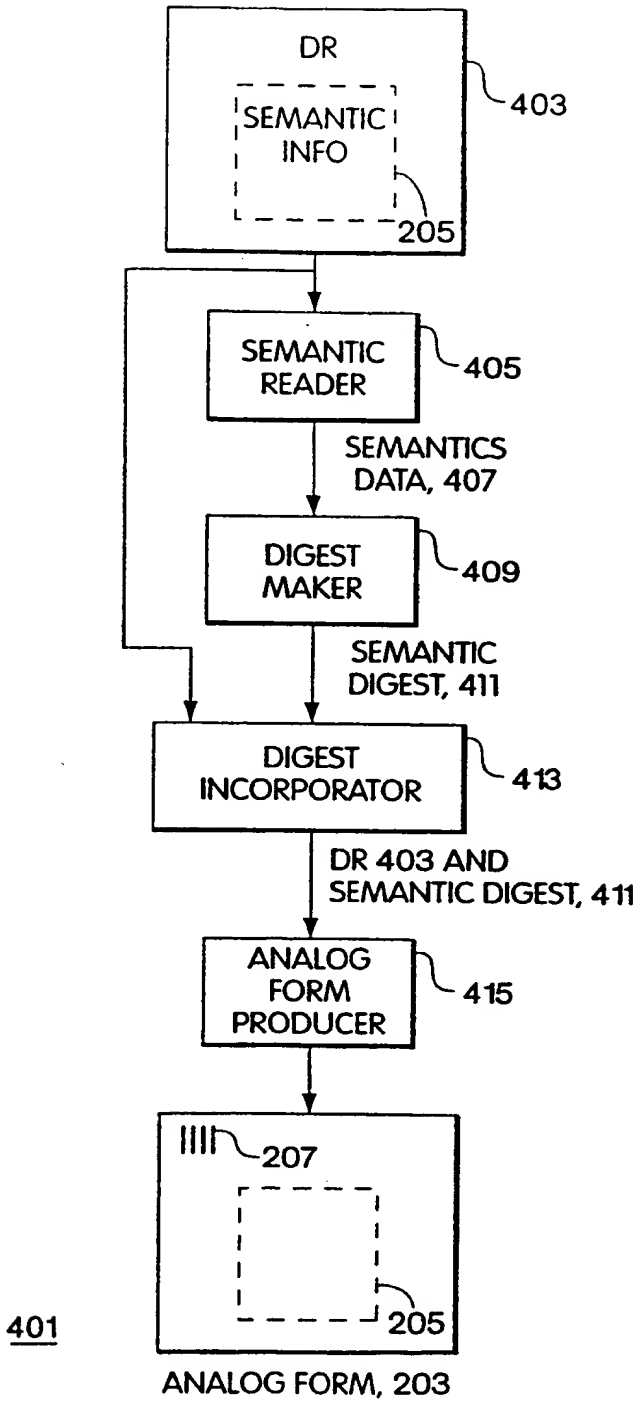


Fig. 4

BNSDOCID: <WO___9957623A2_I_>

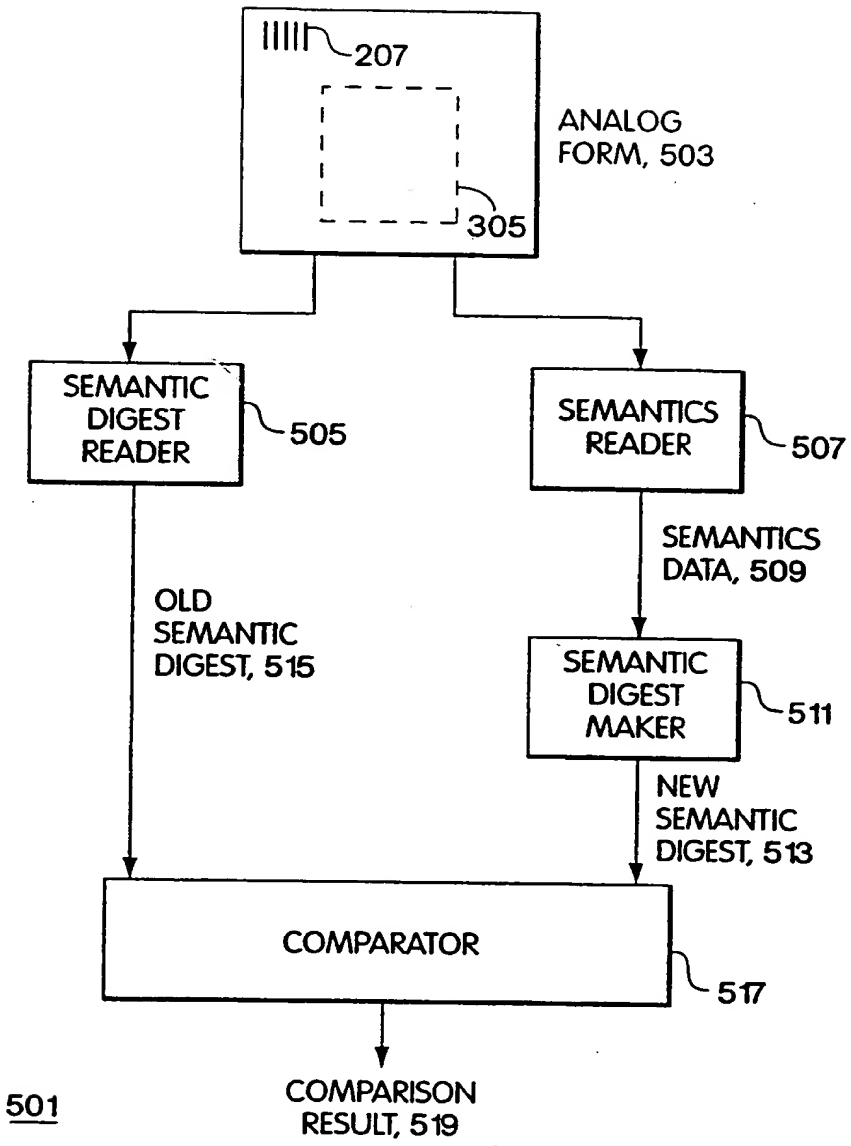


Fig. 5

BNSDOCID: <WO___9957623A2_I_>

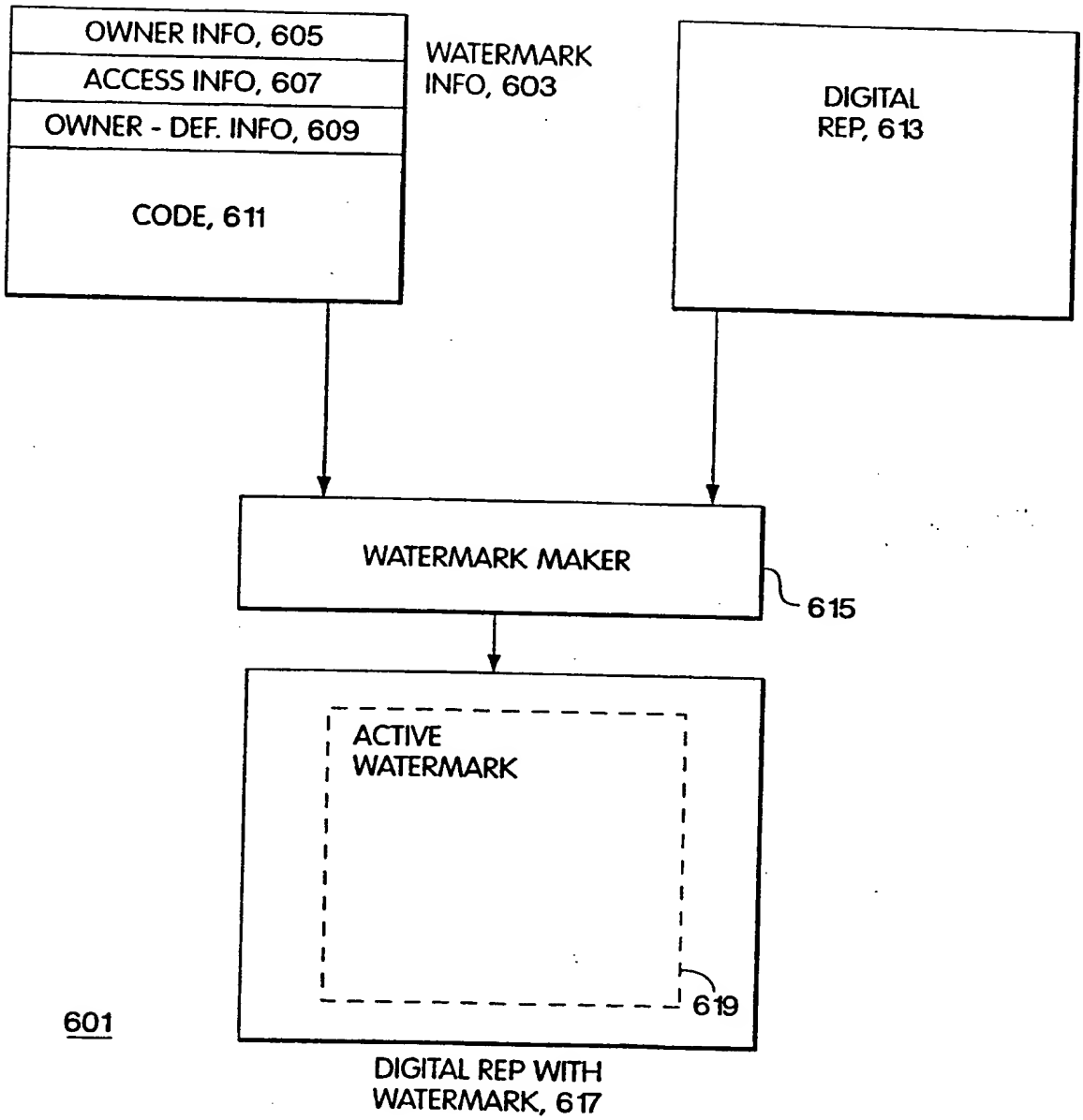


Fig. 6

BNSDOCID: <WO___9957623A2_I_>

```
DatagramSocket s = new DatagramSocket (0);                    703

InetAddress a =                                    705
    netAddress.getByName ("syscop. crcg.edu");                709

DatagramPacket p = new DatagramPacket ('XYZ
    Displayed", 13, a, 14);                                   1715
                  1711          1713
s.send(p);
      1719
```

701

Fig. 7

BNSDOCID: <WO _ 9957623A2_I_>

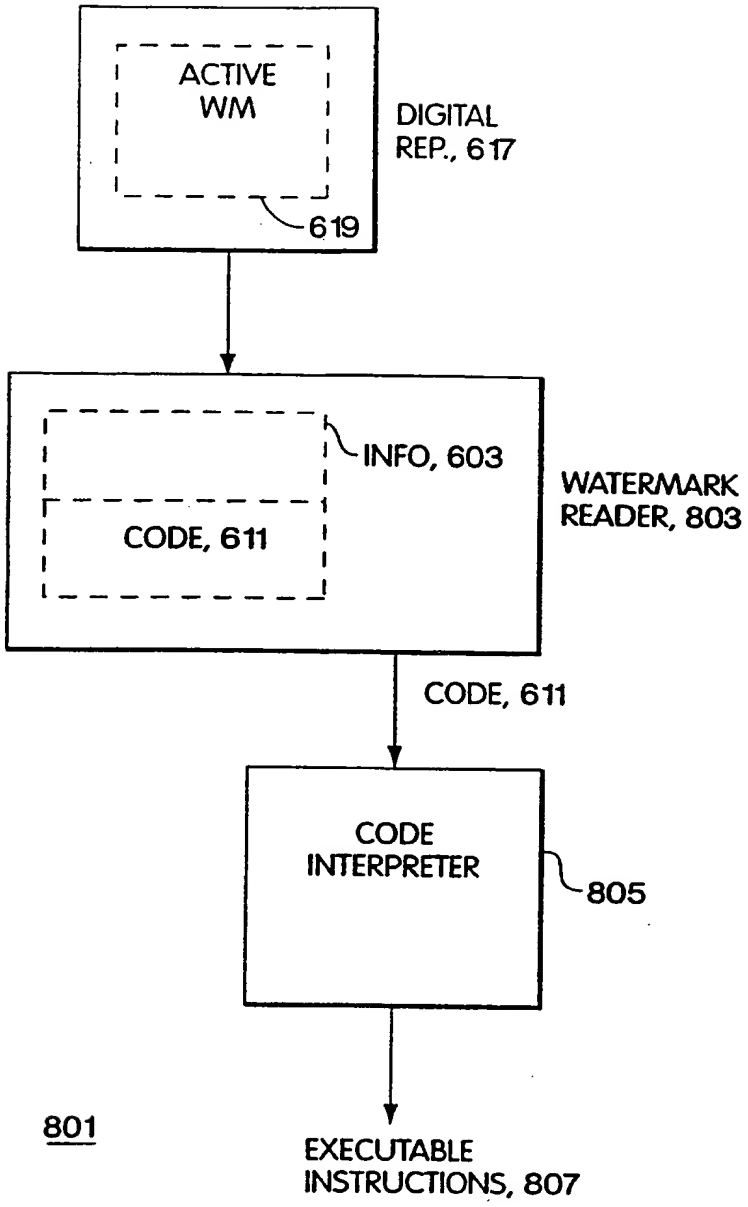


Fig. 8

BNSDOCID: <WO___9957623A2_I_>

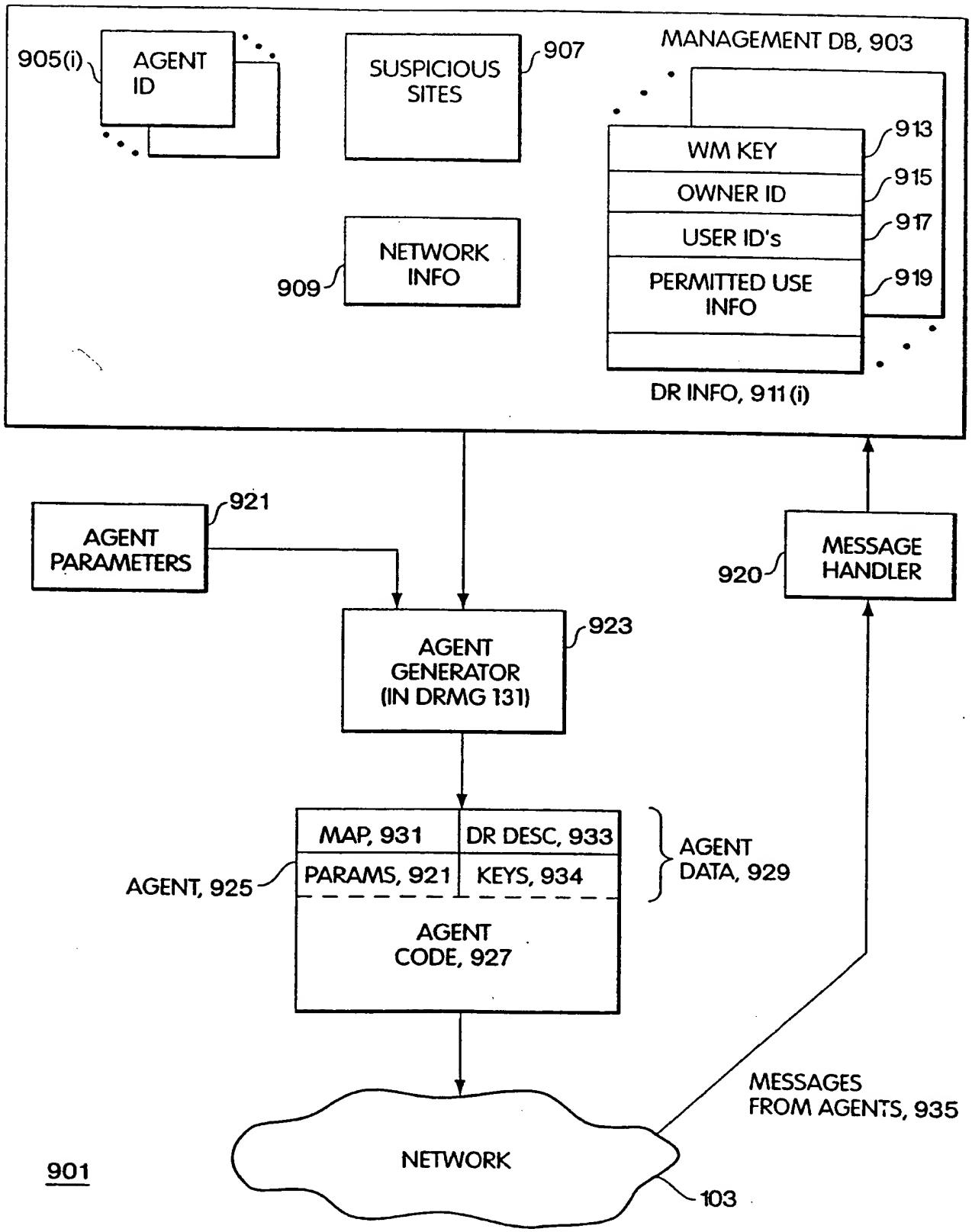


Fig. 9

BNSDOCID: <WO 9957623A2 I >

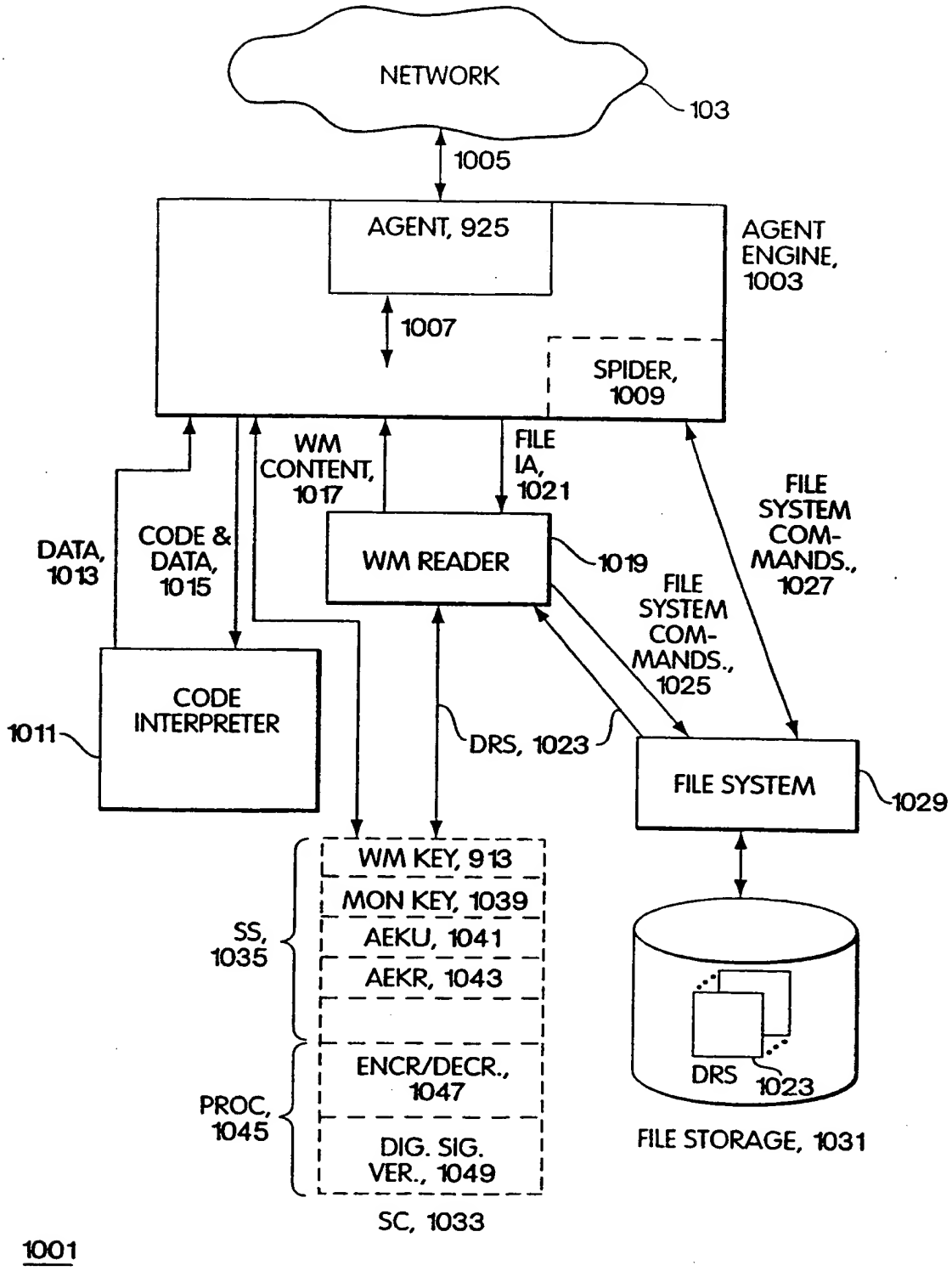


Fig. 10

BNSDOCID: <WO___9957623A2_I_>

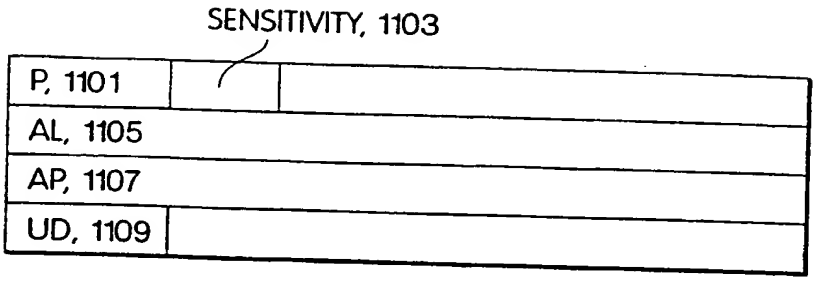


Fig. 11

```
// Instantiate a file filter.
FileFilter fileFilter =new FileFilter();                    1205
// filter out all image files from the file system.
String {} filenames = fileFilter.filterImages();            1207
// get host environment information.
EnvInfo env = getEnvInfo();                                 1209
// construct a new vector used to store action results.
Vector results = new Vector();                              1211

for(int i=0; i<filenames.length; i++) {
  // check each image file for watermark.
  String watermark = checkWatermark (filenames[i]);         1215
  if (watermark != null) { // if a watermark is found

     // match the watermark with the host environment.
     String match  = matchEnv(watermark, env);              1219
     // take action according to the matching result.
     String result = takeAction (match);                    1221
     // add the result to the vector.
     results.addElement (result);                           1223
  }
}
return results;                                             1225
```

1203 1213 1217

1201

Fig. 12

BNSDOCID: <WO___9957623A2_I_>

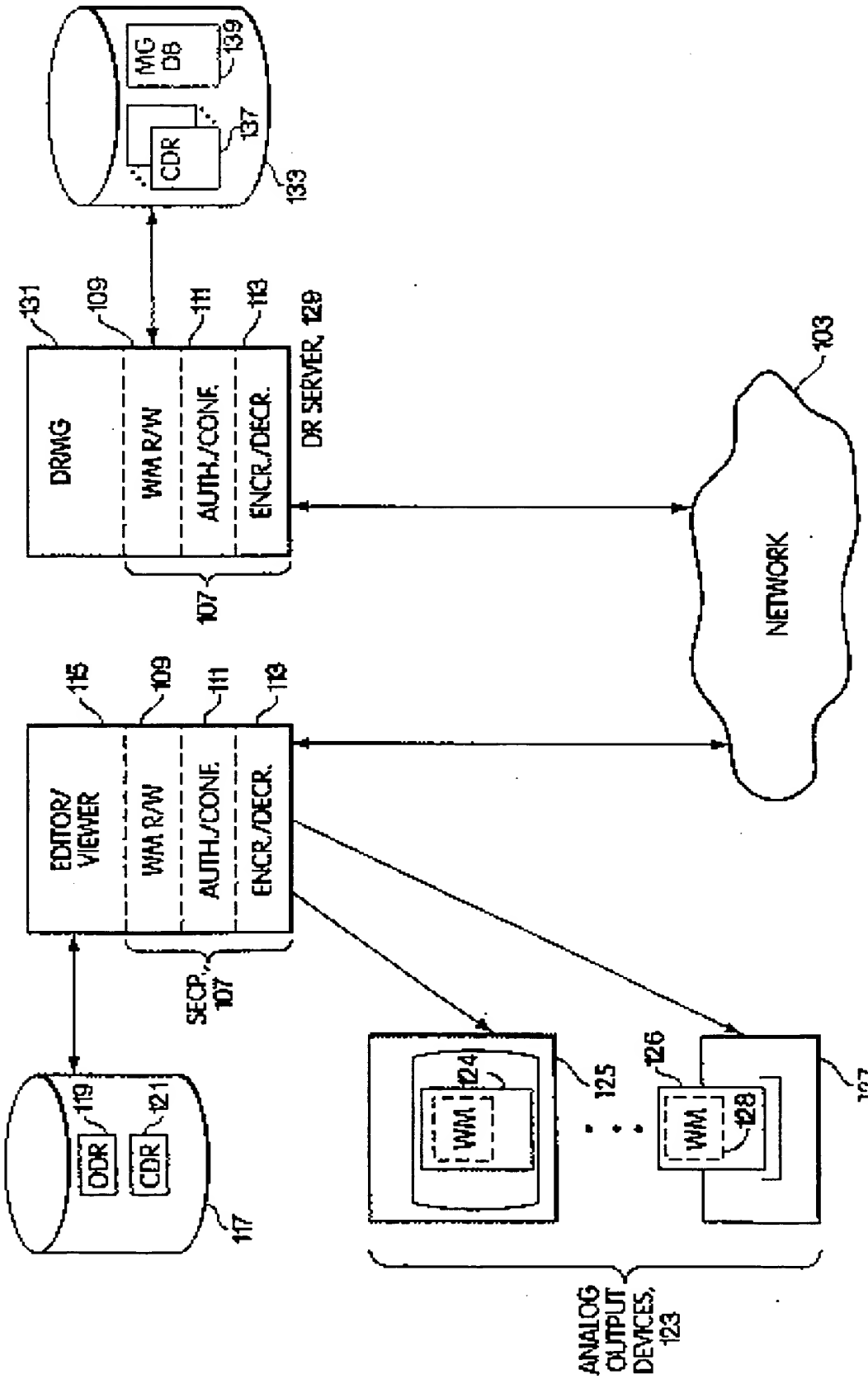


Fig. 1
Prior Art

BNSDOCID: <WO___9957623A2TI_>

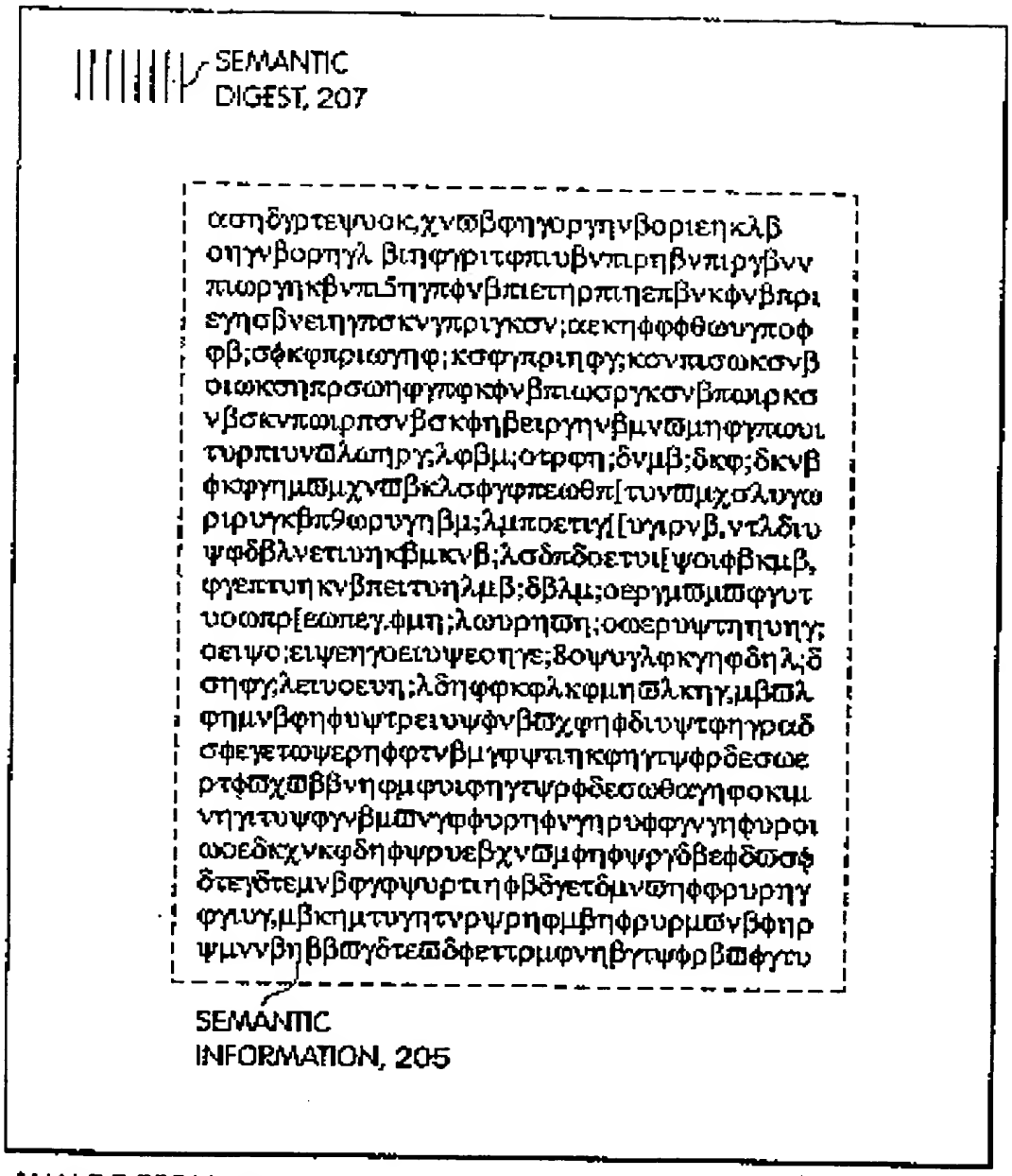


Fig. 2

BNSDOCID: <WO_ 9957623A2TI >

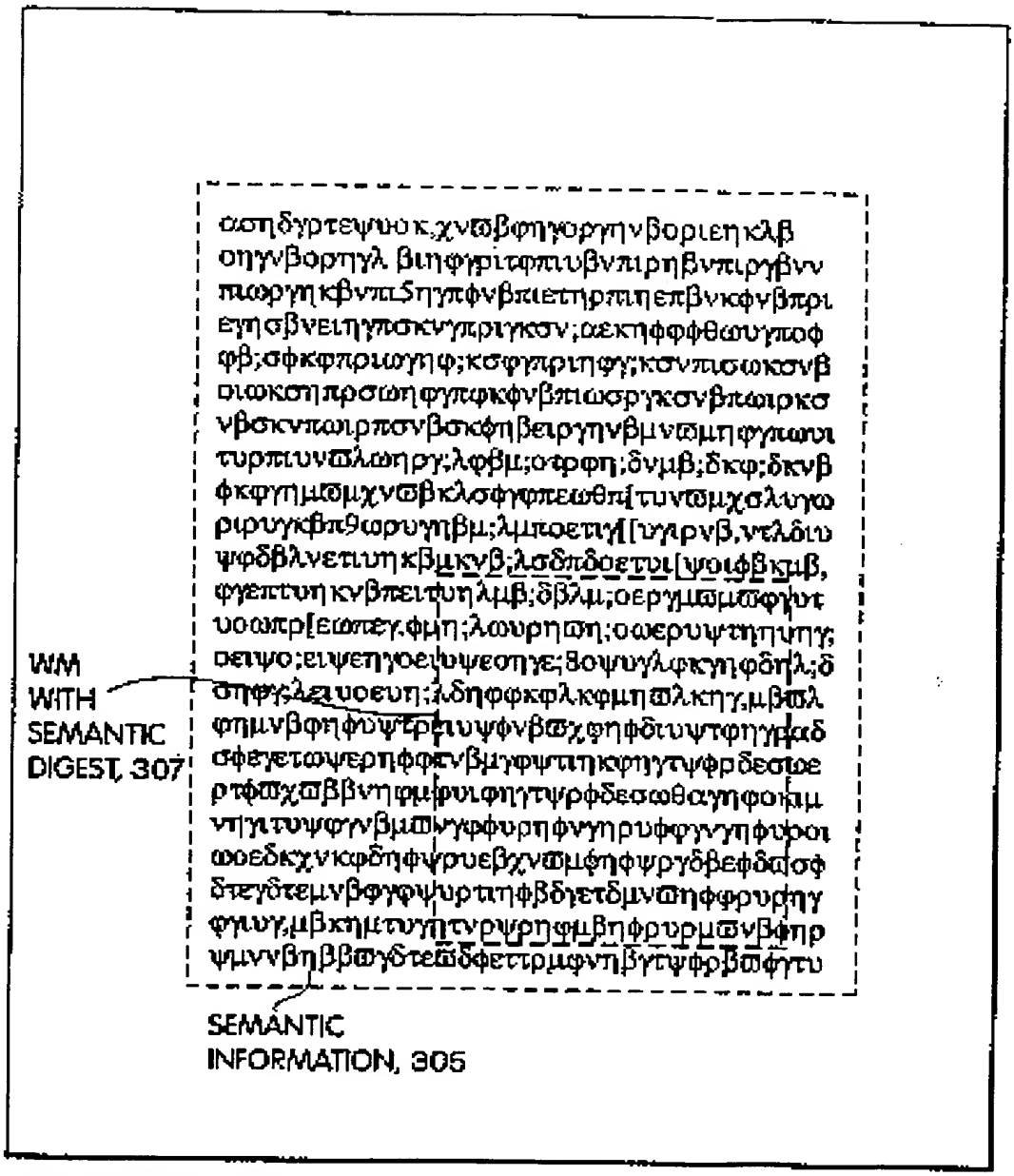


Fig. 3

BNSDOCID: <WO___9957623A2TI_>

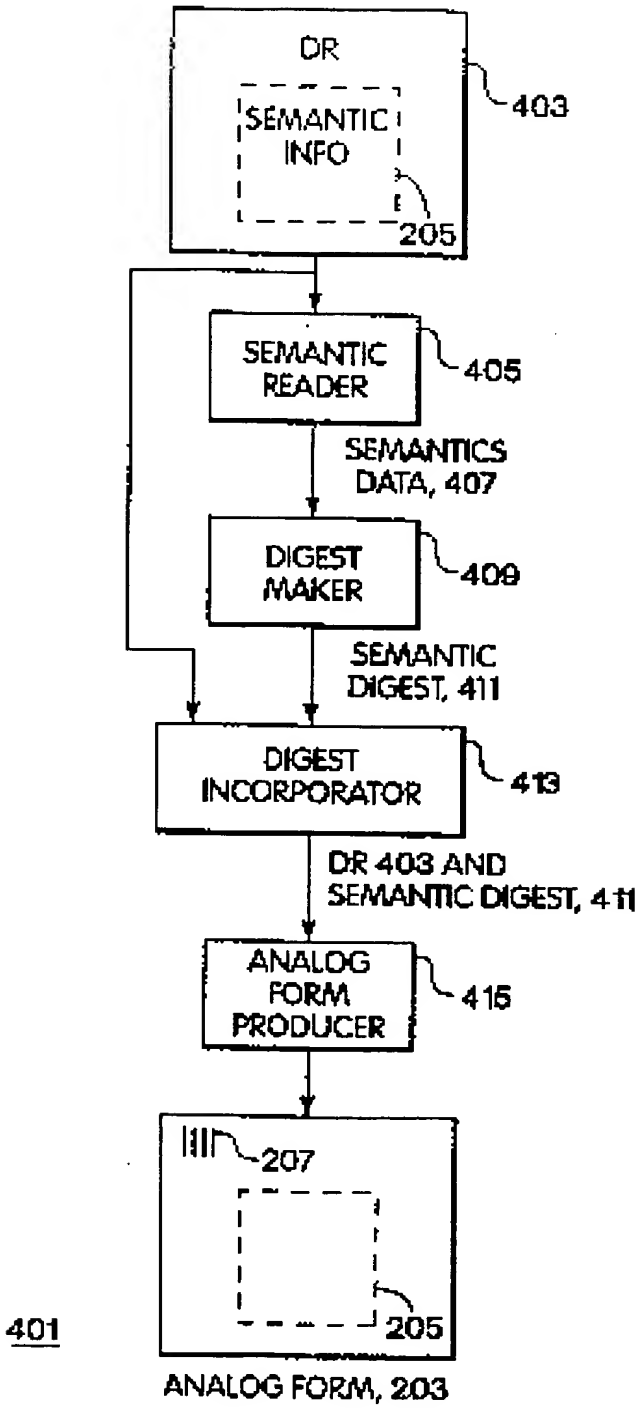


Fig. 4

BNSDOCID: <WO___9957623A2TI_>

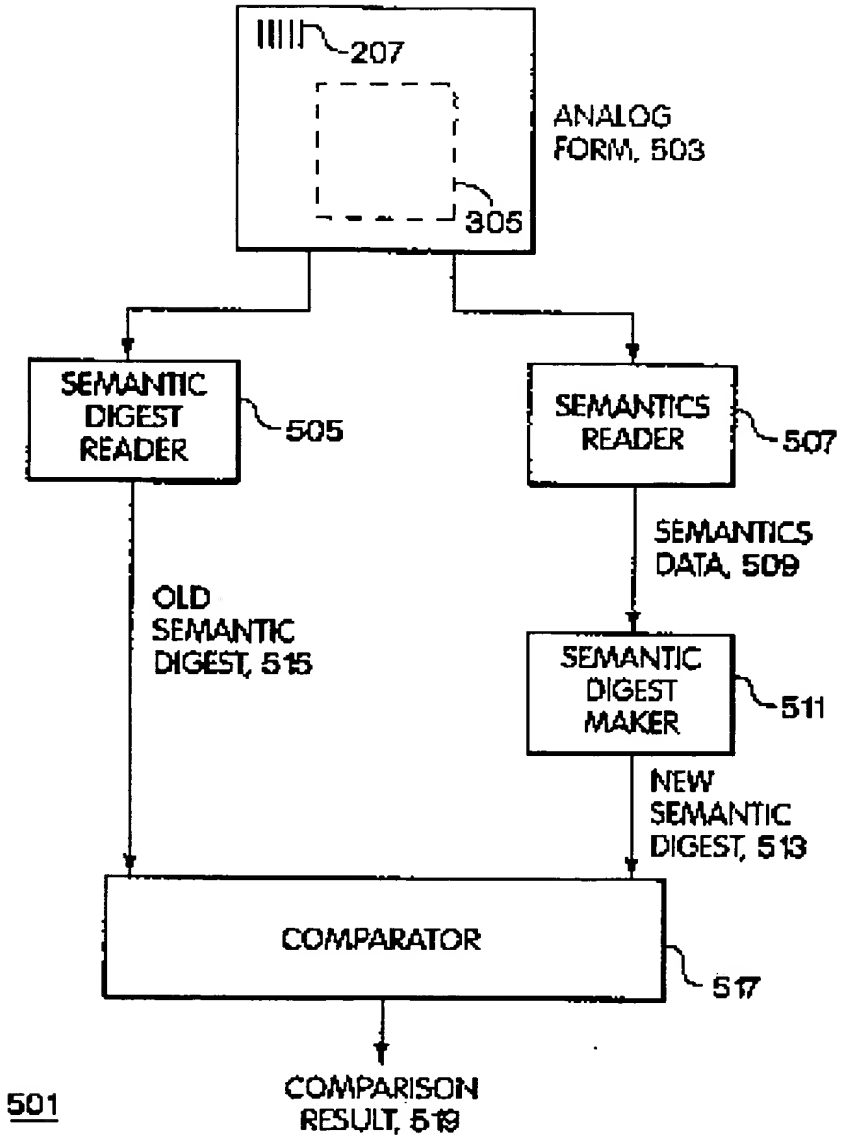


Fig. 5

BNSDOCID: <WO___9957623A2TI_>

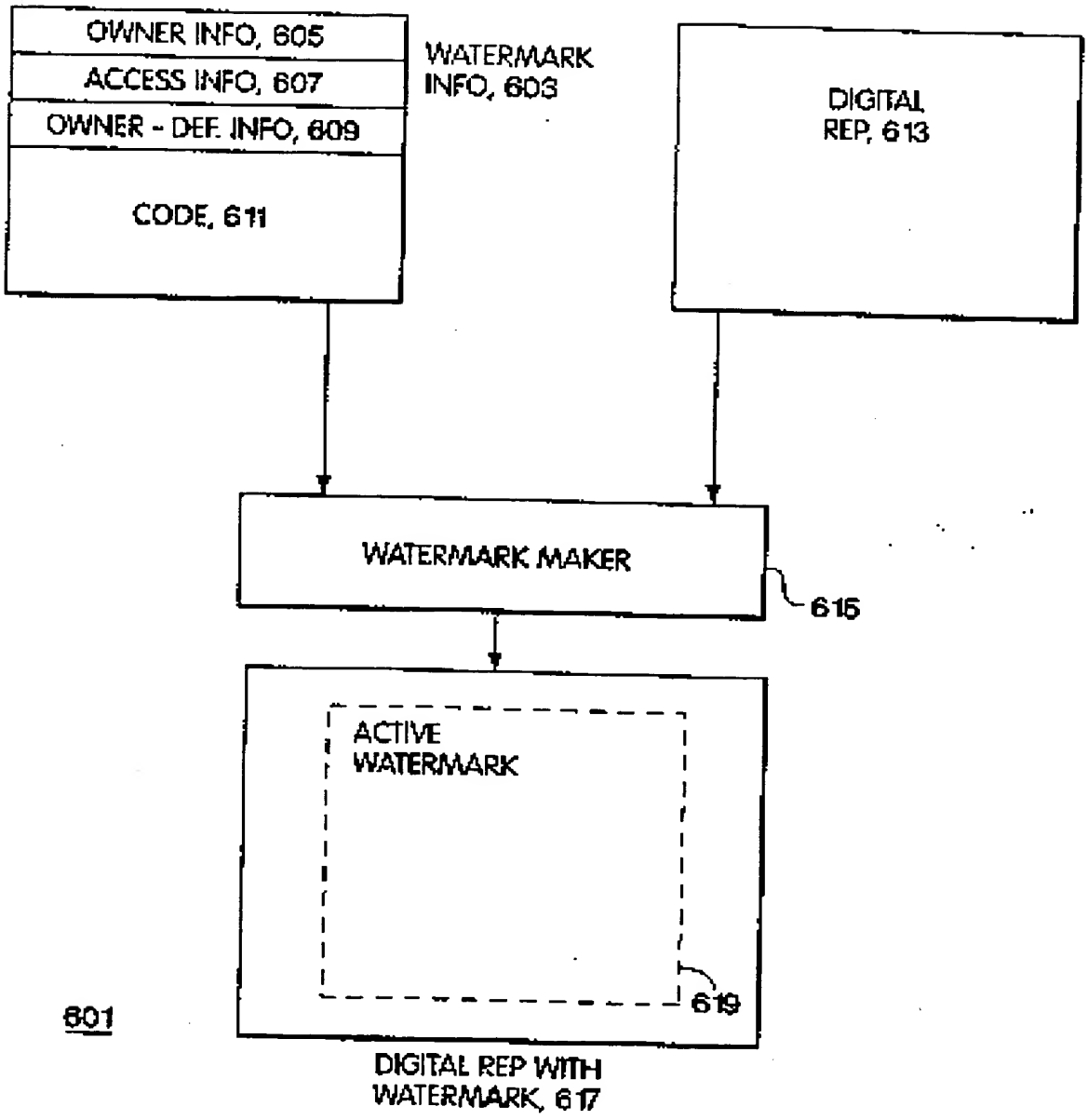


Fig. 6

BNSDOCID: <WO___9957623A2TI_>

```
DatagramSocket s = new DatagramSocket (0);                    703

InetAddress a =                                    705
    netAddress.getByName ("syscop. crcg.edu");                709

DatagramPacket p = new DatagramPacket ('XYZ
    Displayed", 13, a, 14);                                   715
                    711
s.send(p);                    713
        719
```

701

Fig. 7

BNSDOCID: <WO___9957623A2TI_>

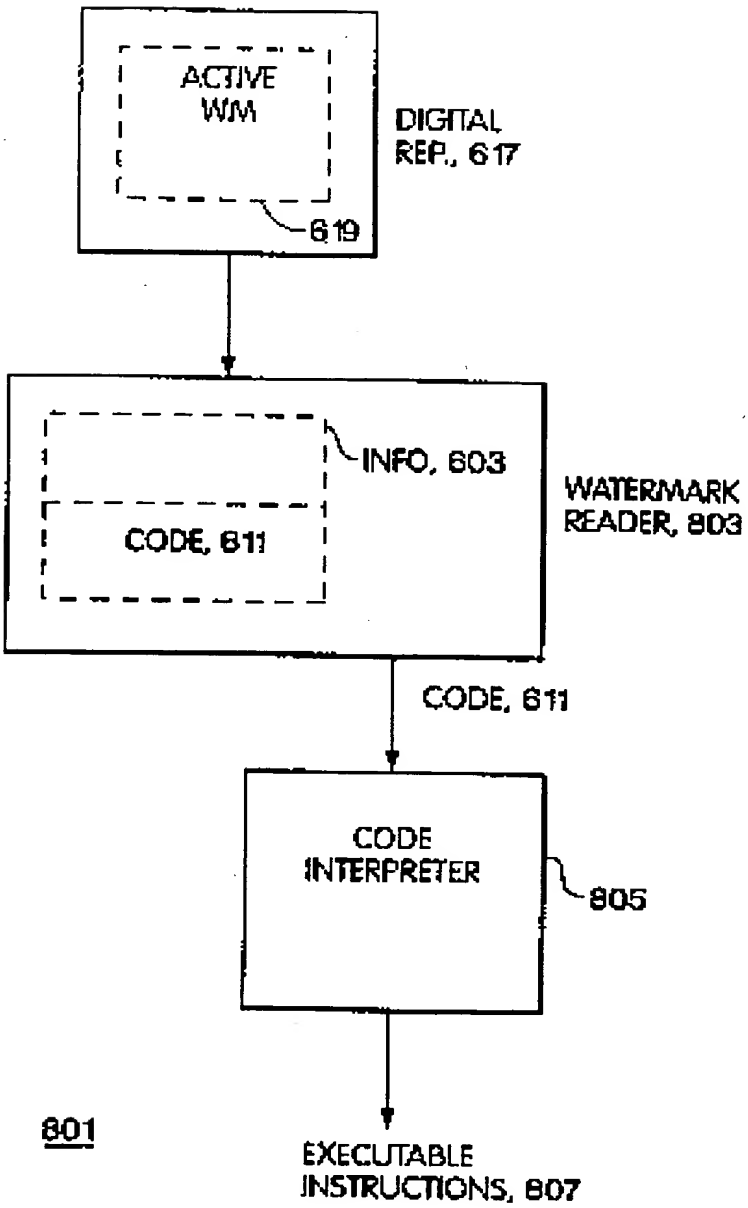


Fig. 8

BNSDOCID: <WO___9957623A2TI_>

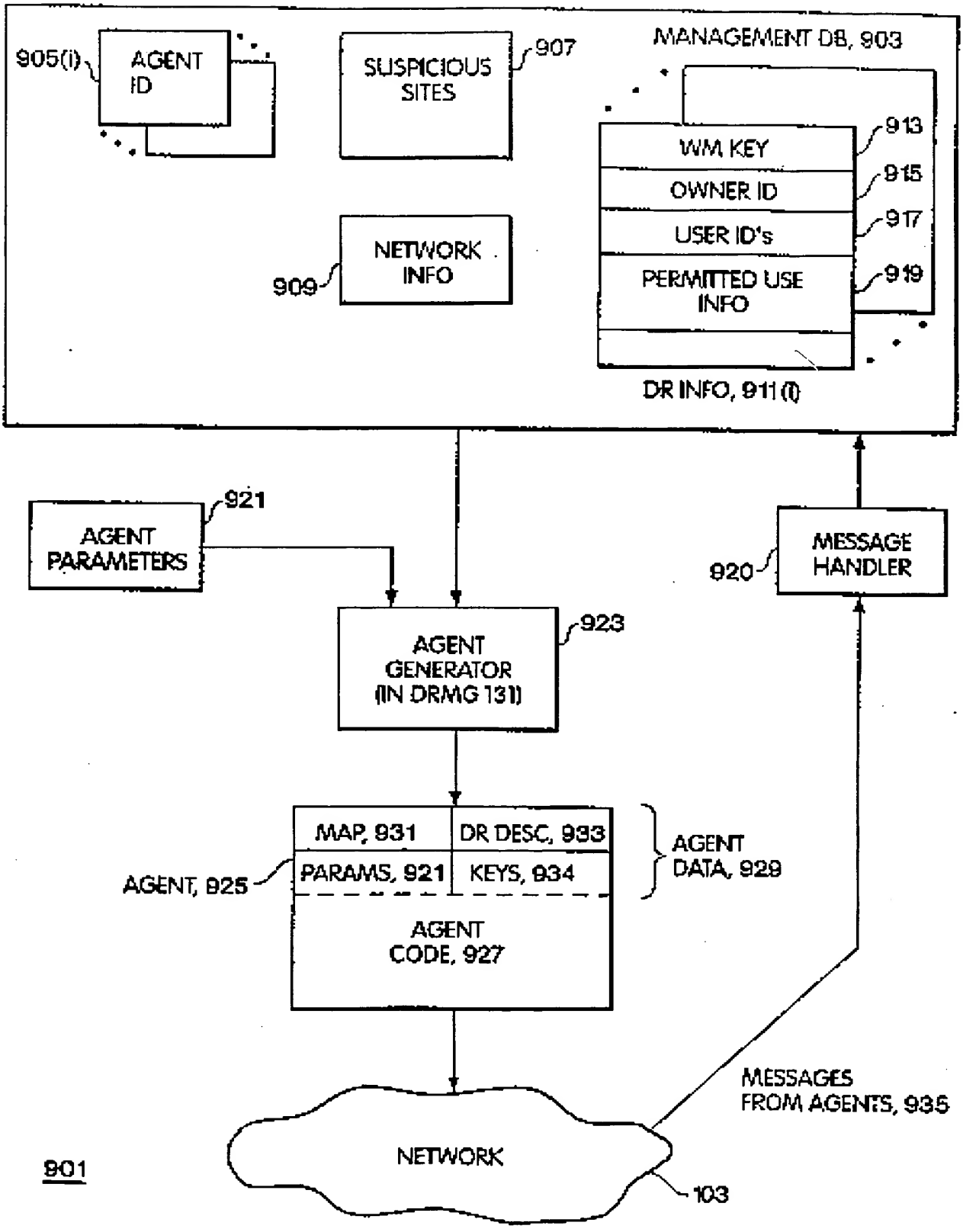


Fig. 9

BNSDOCID: <WO___9957623A2TI_>

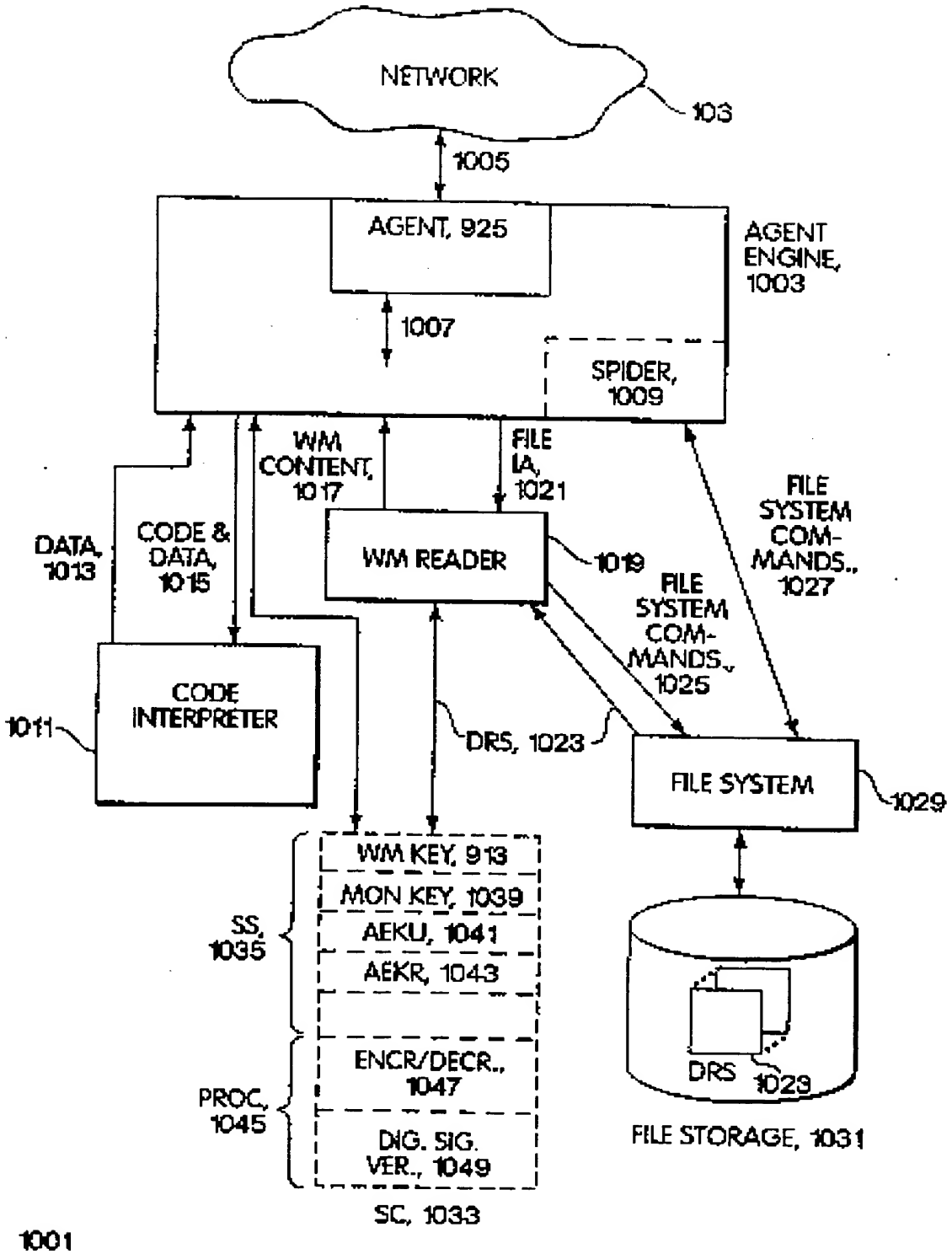


Fig. 10

BNSDOCID: <WO___9957623A2T1_>

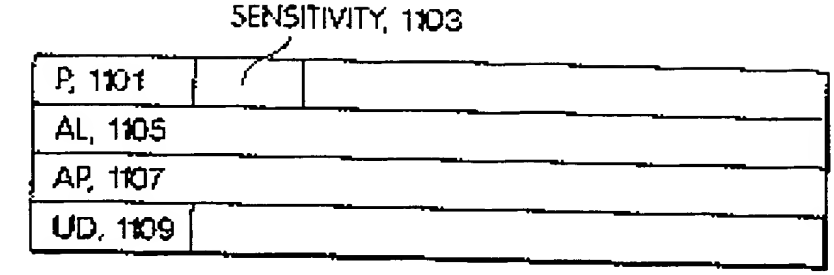


Fig. 11

```
// Instantiate a file filter.
FileFilter fileFilter =new FileFilter();                    1205
// filter out all image files from the file system.
String [] filenames = fileFilter.filterImages();            1207
// get host environment information.
EnvInfo env = getEnvInfo();                                 1209
// construct a new vector used to store action results.
Vector results = new Vector();                              1211

for(int i=0; i<filenames.length; i++) {
  // check each image file for watermark.
  String watermark = checkWatermark (filenames[i]);         1215
  if (watermark != null) { // if a watermark is found

     // match the watermark with the host environment.
     String match  = matchEnv(watermark, env);              1219
     // take action according to the matching result.
     String result = takeAction (match);                    1221
     // add the result to the vector.
     results.addElement (result);                           1223
  }
}
return results;                                             1225
```

1201

1203 1213 1217

Fig. 12

BNSDOCID: <WO___9957623A2TI_>

THIS PAGE BLANK (USPTO)

PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 :
H04N 1/32

A3

(11) International Publication Number: WO 99/57623

(43) International Publication Date: 11 November 1999 (11.11.99)

(21) International Application Number: PCT/IB99/01250

(22) International Filing Date: 29 April 1999 (29.04.99)

(30) Priority Data:
09/070,597 30 April 1998 (30.04.98) US

(71) Applicant: FRAUNHOFER CENTER FOR RESEARCH IN COMPUTER GRAPHICS, INC. [US/US]; Suite 2, 321 S. Main Street, Providence, RI 02903 (US).

(72) Inventor: ZHAO, Jian; 64 Thomas Olney Common, Providence, RI 02904 (US).

(81) Designated States: JP, KR, European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

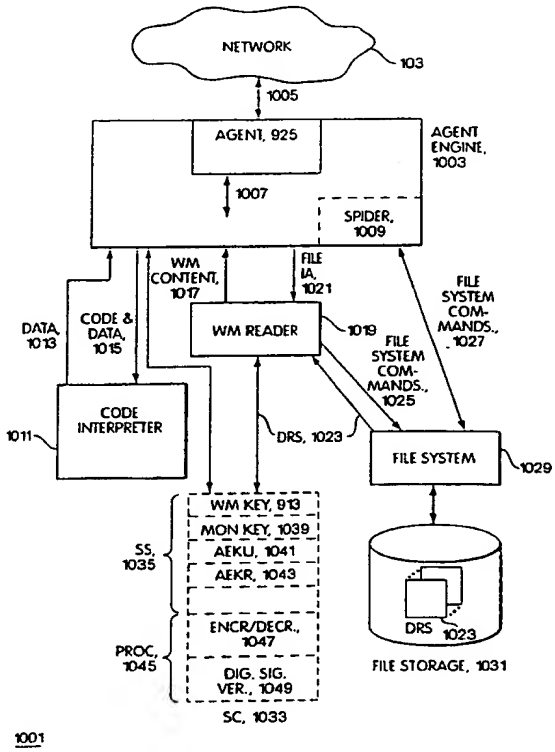
Published
With international search report.
Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

(88) Date of publication of the international search report:
20 April 2000 (20.04.00)

(54) Title: ACTIVE WATERMARKS AND WATERMARK AGENTS

(57) Abstract

Techniques for protecting the security of digital representations, and of analog forms made from them. The techniques include authentication techniques that can authenticate both a digital representation and an analog form produced from the digital representation, an active watermark that contains program code that may be executed when the watermark is read, and a watermark agent that reads watermarks and sends messages with information concerning the digital representations that contain the watermarks. The authentication techniques use semantic information to produce authentication information. Both the semantic information and the authentication information survive when an analog form is produced from the digital representation. In one embodiment, the semantic information is alphanumeric characters and the authentication information is either contained in a watermark embedded in the digital representation or expressed as a bar code. With the active watermark, the watermark includes program code. When a watermark reader reads the watermark, it may cause the program code to be executed. One application of active watermarks is making documents that send messages when they are operated on. A watermark agent may be either a permanent resident of a node in a network or of a device such as a copier or it may move from one network node to another. In the device or node, the watermark agent executes code which examines digital representations residing in the node or device for watermarked digital representations that are of interest to the watermark agent. The watermark agent then sends messages which report the results of its examination of the digital representations. If the watermarks are active, the agent and the active watermark may cooperate and the agent may cause some or all of the code that an active watermark contains to be executed.



BNSDOCID: <WO___9957623A3_I_>

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	**ES**	Spain	**LS**	Lesotho	**SI**	Slovenia
AM	Armenia	**FI**	Finland	**LT**	Lithuania	**SK**	Slovakia
AT	Austria	**FR**	France	**LU**	Luxembourg	**SN**	Senegal
AU	Australia	**GA**	Gabon	**LV**	Latvia	**SZ**	Swaziland
AZ	Azerbaijan	**GB**	United Kingdom	**MC**	Monaco	**TD**	Chad
BA	Bosnia and Herzegovina	**GE**	Georgia	**MD**	Republic of Moldova	**TG**	Togo
BB	Barbados	**GH**	Ghana	**MG**	Madagascar	**TJ**	Tajikistan
BE	Belgium	**GN**	Guinea	**MK**	The former Yugoslav Republic of Macedonia	**TM**	Turkmenistan
BF	Burkina Faso	**GR**	Greece			**TR**	Turkey
BG	Bulgaria	**HU**	Hungary	**ML**	Mali	**TT**	Trinidad and Tobago
BJ	Benin	**IE**	Ireland	**MN**	Mongolia	**UA**	Ukraine
BR	Brazil	**IL**	Israel	**MR**	Mauritania	**UG**	Uganda
BY	Belarus	**IS**	Iceland	**MW**	Malawi	**US**	United States of America
CA	Canada	**IT**	Italy	**MX**	Mexico	**UZ**	Uzbekistan
CF	Central African Republic	**JP**	Japan	**NE**	Niger	**VN**	Viet Nam
CG	Congo	**KE**	Kenya	**NL**	Netherlands	**YU**	Yugoslavia
CH	Switzerland	**KG**	Kyrgyzstan	**NO**	Norway	**ZW**	Zimbabwe
CI	Côte d'Ivoire	**KP**	Democratic People's Republic of Korea	**NZ**	New Zealand		
CM	Cameroon			**PL**	Poland		
CN	China	**KR**	Republic of Korea	**PT**	Portugal		
CU	Cuba	**KZ**	Kazakstan	**RO**	Romania		
CZ	Czech Republic	**LC**	Saint Lucia	**RU**	Russian Federation		
DE	Germany	**LI**	Liechtenstein	**SD**	Sudan		
DK	Denmark	**LK**	Sri Lanka	**SE**	Sweden		
EE	Estonia	**LR**	Liberia	**SG**	Singapore		

BNSDOCID: <WO___9957623A3_I_>

INTERNATIONAL SEARCH REPORT

Int ional Application No
PCT/IB 99/01250

A. CLASSIFICATION OF SUBJECT MATTER
IPC 6 H04N1/32

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 6 H04N

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	BENDER W ET AL: "TECHNIQUES FOR DATA HIDING" IBM SYSTEMS JOURNAL, vol. 35, no. 3/04, 1 January 1996 (1996-01-01), pages 313-335, XP000635079 ISSN: 0018-8670 abstract page 313 -page 314 page 322 page 334 -page 335	1,2,5,9, 10,12, 16,25, 27-29
Y		3,4,11, 20
A		32-58

-/--

[X] Further documents are listed in the continuation of box C.

[X] Patent family members are listed in annex.

° Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
4 February 2000	25. 02. 2000

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Stoffers, C

Form PCT/ISA/210 (second sheet) (July 1992)

BNSDOCID: <WO___9957623A3_I_>

INTERNATIONAL SEARCH REPORT

Inte ional Application No
PCT/IB 99/01250

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document, with indication,where appropriate, of the relevant passages	Relevant to claim No.
P,X	US 5 832 119 A (RHOADS GEOFFREY B) 3 November 1998 (1998-11-03) abstract; figure 27 column 1, line 19 - line 30 column 1, line 48 - line 52 column 2, line 7 - line 10 column 3, line 57 - line 60 column 53, line 63 -column 54, line 5 column 56, line 48 - line 54 claims 1,7-10,22,23,26-28,44,46.47,56,57	1,5-20, 25-29
Y	EP 0 798 892 A (IBM) 1 October 1997 (1997-10-01) abstract figure 6 column 1, line 38 - line 41	3,4,11, 20
A		6-8, 13-15
A	US 5 710 834 A (RHOADS GEOFFREY B) 20 January 1998 (1998-01-20) cited in the application abstract column 1, line 39 - line 57 column 2, line 13 - line 24 column 3, line 34 - line 41 column 4, line 66 -column 5, line 2 column 53, line 49 - line 55 column 54, line 31 - line 40 column 55, line 10 - line 19 column 67, line 13 - line 19 figures 2,3,5,13,17,22-24 claims 1,4-6	1-21, 25-31
A	ZHAO J: "LOOK, IT'S NOT THERE" BYTE, vol. 22, no. 1, 1 January 1997 (1997-01-01), page 7/8, 10, 12 XP000199953 ISSN: 0360-5280 cited in the application the whole document	1-21, 25-31
A	CHOUDHURY A K ET AL: "COPYRIGHT PROTECTION FOR ELECTRONIC PUBLISHING OVER COMPUTER NETWORKS" IEEE NETWORK: THE MAGAZINE OF COMPUTER COMMUNICATIONS, vol. 9, no. 3, 1 May 1995 (1995-05-01), pages 12-20, XP000505280 ISSN: 0890-8044 the whole document	1-21, 25-31

-/--

6

BNSDOCID: <WO___9957623A3_I_>

INTERNATIONAL SEARCH REPORT

Int tional Application No
PCT/IB 99/01250

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category ³	Citation of document, with indication,where appropriate. of the relevant passages	Relevant to claim No.
A	EP 0 747 843 A (IBM) 11 December 1996 (1996-12-11) abstract column 5, line 53 -column 6, line 4 column 6, line 39 - line 43 column 7, line 25 -column 8, line 10 column 18, line 54 - line 31 claim 21	32-58
A	WO 97 34426 A (ENCANTO NETWORK INC) 18 September 1997 (1997-09-18) abstract page 7, line 20 - line 23 figure 2 page 8, line 1 - line 31 page 10, line 6 -page 11, line 15 claims 20,22-27	32-58
A	WO 97 43736 A (RHOADS GEOFFREY B ;DIGIMARC CORP (US)) 20 November 1997 (1997-11-20) page 52, line 28 -page 53, line 5 page 77, line 17 - line 21 figure 27	32-58

Form PCT/ISA/210 (continuation of second sheet) (July 1992)

BNSDOCID: <WO___9957623A3_I_>

INTERNATIONAL SEARCH REPORT

Inte. ational application No.
PCT/IB 99/01250

Box I Observations where certain claims were found unsearchable (Continuation of item 1 of first sheet)

This International Search Report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. ☐ Claims Nos.:
because they relate to subject matter not required to be searched by this Authority, namely:

2. ☐ Claims Nos.:
because they relate to parts of the International Application that do not comply with the prescribed requirements to such an extent that no meaningful International Search can be carried out, specifically:

3. ☐ Claims Nos.:
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

Box II Observations where unity of invention is lacking (Continuation of item 2 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:

see additional sheet

1. ☒ As all required additional search fees were timely paid by the applicant, this International Search Report covers all searchable claims.

2. ☐ As all searchable claims could be searched without effort justifying an additional fee, this Authority did not invite payment of any additional fee.

3. ☐ As only some of the required additional search fees were timely paid by the applicant, this International Search Report covers only those claims for which fees were paid, specifically claims Nos.:

4. ☐ No required additional search fees were timely paid by the applicant. Consequently, this International Search Report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:

Remark on Protest

☐ The additional search fees were accompanied by the applicant's protest.

☒ No protest accompanied the payment of additional search fees.

BNSDOCID: <WO___9957623A3_I_>

FURTHER INFORMATION CONTINUED FROM PCT/ISA/ 210

This International Searching Authority found multiple (groups of) inventions in this international application, as follows:

1. Claims: 1-31

Digital watermark including program code whereby an apparatus that reads the digital watermark may cause the program code to be executed.

2. Claims: 32-58

a method of obtaining information about digital representations that contain watermarks, the method being employed in a network and comprising the steps of sending an agent that includes a program that reads watermarks via the network to a system on which digital representations with watermarks my be resident; and receiving a message from the agent program via the network indicating results of the agent program`s reading of watermark in the system.

BNSDOCID: <WO___9957623A3_I_>

INTERNATIONAL SEARCH REPORT

Information on patent family members

Int. tional Application No
PCT/IB 99/01250

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 5832119 A	03-11-1998	US 5841978 A	24-11-1998
		US 5636292 A	03-06-1997
		US 5768426 A	16-06-1998
		AU 6022396 A	29-11-1996
		CA 2218957 A	14-11-1996
		EP 0824821 A	25-02-1998
		WO 9636163 A	14-11-1996
		US 5862260 A	19-01-1999
		EP 0737387 A	16-10-1996
		JP 9509795 T	30-09-1997
		US 5841886 A	24-11-1998
		CA 2174413 A	26-05-1995
		EP 0959620 A	24-11-1999
		EP 0959621 A	24-11-1999
		WO 9514289 A	26-05-1995
		US 5748763 A	05-05-1998
		US 5850481 A	15-12-1998
		US 5745604 A	28-04-1998
EP 0798892 A	01-10-1997	US 5673316 A	30-09-1997
		JP 10040100 A	13-02-1998
US 5710834 A	20-01-1998	US 5841978 A	24-11-1998
		US 5841886 A	24-11-1998
EP 0747843 A	11-12-1996	US 5701451 A	23-12-1997
		JP 9114863 A	02-05-1997
		US 5974441 A	26-10-1999
WO 9734426 A	18-09-1997	AU 1972597 A	01-10-1997
WO 9743736 A	20-11-1997	US 5862260 A	19-01-1999
		AU 3008697 A	05-12-1997

BNSDOCID: <WO___9957623A3_I_>